Frontiers in Computing and Intelligent Systems
ISSN: 2832-6024 | Vol. 12, No. 1, 2025

A Method for Redirecting Humanoid Robots Based on Segmented Geometric Inverse Kinematics

Haotian Wang [a], Zhongxue Gan *

Academy for Engineering and Technology, Fudan University, Shanghai, 200433, China
*** Corresponding author:** Zhongxue Gan (Email: ganzhongxue@fudan.edu.cn), [a] 22210860060@m.fudan.edu.cn

Abstract: The current vision-based dexterous teleoperation system for robotic upper limbs mainly focuses on the recognition and reconstruction of human hand and finger postures, while the consideration of the overall arm postures is relatively insufficient, resulting in a non-negligible accumulation of errors in specific operation scenarios. To address this problem, this study proposes a novel teleoperation framework based on the human skeleton tree topology. The method captures the operator's complete upper limb posture information in real time through optical sensing devices, maps the human body motion to the robot joint space using an optimised skeleton reorientation algorithm, and achieves closed-loop feedback through a stereo vision system. The experimental results show that the proposed method has significant improvement in motion acquisition efficiency, mapping accuracy and system robustness, and provides reliable technical support for accurate teleoperation in complex environments. This study provides a new research idea and implementation way for human-robot collaboration in robot upper limb teleoperation.

Keywords: Humanoid Robot; Motion Reorientation; Inverse Kinematics; Teleoperation.

1. Introduction

Remote teleoperated robotic technology shows a broad application prospect in high-precision operation fields such as hazardous environment operation, medical surgery and space exploration. Traditional teleoperation systems usually rely on specialized control equipment, which is not only complicated to operate, but also requires specialized training, limiting the popularity of the application. Although vision-based intuitive teleoperation methods provide a “what you see is what you get” control mode, most of the existing systems focus on the hand posture and ignore the whole upper limb, and when dealing with the differences in human-computer structure, it is difficult to accurately map the complex three-dimensional operation trajectory by simple scaling, which leads to the accumulation of errors. Aiming at these problems, this paper proposes a teleoperation scheme based on human skeleton tree reorientation, which collects the operator's complete upper limb posture information through a high-precision visual sensing system, establishes a human-robot mapping model considering structural differences and kinematic constraints, and designs a real-time optimization algorithm to ensure the accuracy of the motion while taking into account the operability and stability of the system, so as to enhance the efficiency and quality of the operation in the complex tasks. The system is optimized in real time to ensure the accuracy of the motion while taking into account the maneuverability and stability of the system, so as to improve the efficiency and quality of operation in complex tasks.

2. Related Work

In this paper, we use the SMPL (Skinned Multi-Person Linear Model) format to save human whole-body motion information, which is a widely used parametric human model proposed by Loper et al. in 2015.SMPL captures the complex morphology and motion of the human body through a small number of parameters, which mainly contains two types of parameters: shape parameters and posture parameters. Shape parameters are extracted from real human body scanning data through principal component analysis, describing body features such as height, weight, and limb length; posture parameters describe the rotation information of 24 joints (23 joints in the whole body plus 1 joint representing global posture, each with 3 degrees of freedom), and accurately represent various postures of the human body through the angles of the joints, as shown in Fig. 1. By virtue of its ability to efficiently represent the shape and posture of the human body with its ability to efficiently represent human shapes and postures, SMPL is widely used in humanoid robotics, motion capture and virtual reality.

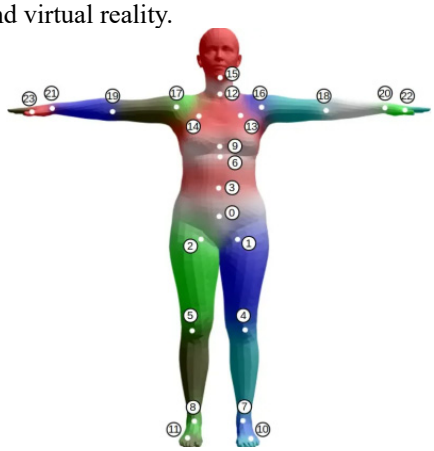

Fig 1. SMPL joint order labeling diagram

3. Upper Limb Teleoperation for Robots Based on Human Skeletal Tree Reorientation

3.1. Overall System Framework

The robot upper limb teleoperation system based on skeleton tree reorientation proposed in this study mainly contains three core modules: human posture recognition module, skeleton tree reorientation module and robot control module. The system collects human upper limb movements through the

camera, converts them into standardized SMPL human model parameters through WHAM (Whole-Body Human Action Model) algorithm, and then maps the human movements to the robot joint space based on the skeleton tree reorientation algorithm, and finally drives the robot to perform the corresponding actions through the control module. At the same time, the binocular camera on the robot's head provides real-time feedback of the environment to the operator, forming a complete closed-loop control. The system architecture is shown in Figure 2.

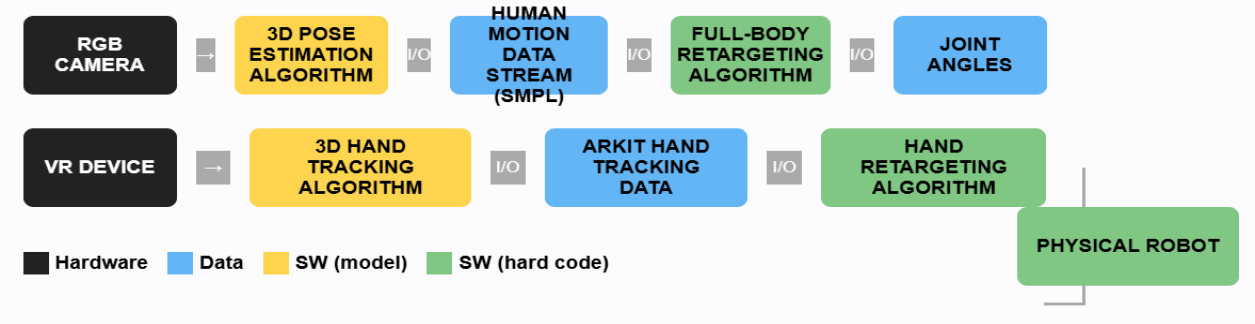

Fig 2. Overall algorithm structure

3.2. Human Skeleton Tree Construction and Mapping

3.2.1. Skeleton Tree Construction

In order to realize the accurate retargeting of upper limb movements, it is first necessary to build a human skeleton tree model with the parameters of the robot structure. Unlike traditional methods that focus only on the end-effector, this study accurately measures and models the relative positions of each joint node of the robot. As shown in Table 1, in particular, specific localization methods are used for different types of joint nodes:

1. for nodes driven by three motors, such as shoulders and wrists, the intersection of the rotation axes of the three motors is selected as the center of rotation

2. For nodes driven by a single motor, such as the elbow, the intersection of the line between the centers of the upper and lower nodes and the axis of the motor is selected as the center of rotation.

Table 1. Logical parent-child relationship mapping of the nodes in the skeleton tree of Guanghua 1 robot and their initial relative displacement table

Joint	Parent Node	Initial Relative Displacement (Relative to Parent Node)
Root Node	World Coordinate Origin	[0,0,1.20]
Waist Node	Root Node	[0,0,0.133]
Left Shoulder Node	Waist Node	[0,0.240,0.285]
Left Elbow Node	Left Shoulder Node	[0,0.240,0]
Left Wrist Node	Left Elbow Node	[0,0.240,0]
Right Shoulder Node	Waist Node	[0,-0.214,0.285]
Right Elbow Node	Right Shoulder Node	[0,-0.240,0]
Right Wrist Node	Right Elbow Node	[0,-0.240,0]

Establishing the nodes in this way ensures that the skeleton tree is precisely aligned with the individual rigid bodies of the robot. Specifically, the core of this method is to completely match the rotation center of the skeleton with the actual mechanical rotation center of the robot, so as to realize the consistency between the geometric structure and the kinematic behavior, and thus the kinematic model of the skeleton can directly reflect the actual motion state of the robot, avoiding the deviation between the virtual and the real. The effect of the completed human skeleton tree is shown in Figure 3:

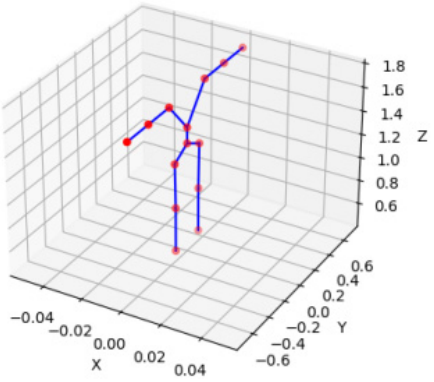

Fig 3. Initial pose of the human skeleton tree of the robot

It is worth noting that, in this paper, when establishing the initial human skeleton tree of the robot, we did not establish it according to the default zero position of the robot, i.e., the posture of the arms hanging down, but rather, we established it according to the default posture of SMPL, i.e., the arms are flat and extended as the initial posture. This facilitates the subsequent computation of the skeleton tree and avoids the need to add additional rotation matrices at the shoulders of the robot, which in turn increases the computational effort.

3.2.2. Positional Mapping of SMPL Nodes to Human Skeleton Tree Nodes

In this section, the mapping relationship of the rotation matrix between each kinematic node of the robot and the nodes of the SMPL model is elaborated. Specifically, for each joint node in the robot kinematic chain, there is a clear mapping relationship between its rotation matrix and the corresponding node in the SMPL model, which can be described in Table 2 below:

Table 2. Skeleton tree nodes and SMPL rotation node correspondences

Joint	Corresponding SMPL Node	Corresponding Rotation Matrix
Left Shoulder Node	Left Clavicle, Left Shoulder	$R_{16} \cdot R_{12}$
Left Elbow Node	Left Elbow	R_{18}
Left Wrist Node	Left Wrist	R_{20}
Right Shoulder Node	Right Clavicle, Right Shoulder	$R_{17} \cdot R_{14}$
Right Elbow Node	Right Elbow	R_{19}
Right Wrist Node	Right Wrist	R_{21}

The algorithm designs a pose mapping method based on the SMPL human skeleton tree to realize the conversion from SMPL pose data to robot rigid body position. The algorithm processes the SMPL pose sequence frame by frame, first initializing the transfer matrix of the root node, then iterating through each joint node, extracting the parent node information, the initial displacement and rotation matrices, and splicing them to form the transfer matrix. Next, the local transfer matrix of the current node relative to the root node is computed by matrix multiplication, and the local pose of each rigid body end relative to the robot root node is finally generated. These data provide the basis for subsequent motion reorientation and inverse kinematics solving. As shown in Fig. 4, the results of the robot skeleton tree node position visualization based on SMPL mapping are demonstrated.

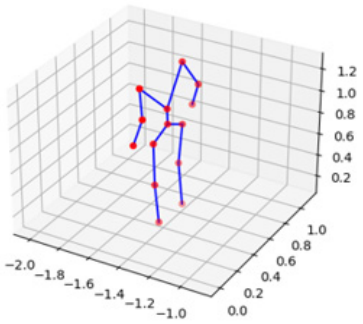

Fig 4. Example of human skeleton tree motion visualization

3.2.3. Human Skeleton Tree Based Reorientation Algorithm

The algorithm realizes the reorientation calculation from the human skeleton tree to the robot joint angles, and the core adopts the inverse kinematics iterative solution method. The algorithm processes the SMPL human pose data sequence, and first converts the SMPL poses into the robot rigid body's position in the local coordinate system, and extracts the key node transfer matrix. Subsequently, iterative optimization is performed: the node poses corresponding to the current robot joint angles are computed by positive kinematics, and complete pose matching (6-dimensional vectors containing displacements and axis angles) is used for end nodes, while only position matching (3-dimensional vectors) is used for non-end nodes. This discretization is based on the fact that the poses of intermediate nodes can be deduced from their relative positions with respect to their parent and child nodes, thus saving computational resources. The algorithm uses the ipopt method to minimize the overall error function until convergence conditions are reached. The algorithm is designed to be adaptable enough to handle the complete SMPL skeleton tree data for full-body coordinated mapping, as well as adapting the work by adjusting the error function in limited input scenarios with only wrist and finger poses to support different motion capture modalities, which improves the flexibility and versatility of the system.

3.3. Vision-based Human Motion Capture System

This study proposes two complementary motion capture approaches: upper limb capture based on WHAM and finger capture based on AR devices.The WHAM system accurately captures shoulder, elbow, and wrist motions, and is suitable for a wide range of upper limb motions, but has limited control of the fingers; while the AR devices capture wrist and finger postures with high precision, and support detailed hand manipulation, but lack monitoring of the elbow and shoulder. When the two are combined, the system provides full-link high-precision control from shoulder to fingertips, which is suitable for complex tasks such as precision grasping, assembly and medical treatment to achieve efficient spatial localization and fine manipulation.

3.3.1. WHAM-based Upper Limb Motion Capture

WHAM is an advanced visual motion capture method capable of reconstructing the 3D pose and shape of the human body from monocular video and outputting SMPL model parameters. Since the original design of WHAM favors recognition quality and ignores time efficiency, it is not suitable for real-time human-computer interaction application scenarios, so this study performs a number of optimizations for the WHAM algorithm to make it suitable for real-time teleoperation scenarios:

Offline process to online process: WHAM is initially designed for offline video human motion capture, and its original process sequentially traverses all the video frames for bounding box selection, then traverses all the frames again to extract 2D keypoints, and finally traverses all the frames to extract 3D SMPL features. In this study, this process is reconstructed and converted into an online video recognition model, where only one bounding box selection, feature extraction and SMPL recognition are performed per iteration. This optimized flow is shown in Figure 5.

2. Model Lightweight Implementation: The WHAM algorithm first uses the Yolo model to detect the human body in the image as a bounding box, and then inputs the detected image regions into the vit-pose model to extract the 2D key points of the human body and jointly inputs this information into the WHAM network system to perform 3D human action recognition. In this paper, by replacing yolov8x and ViTPose-H (the largest variants) used in the original implementation with yolov8s and ViTPose-S (the smallest variants), the processing times that originally required an average of 25.55 ms and 36.95 ms, respectively, are reduced to an average of 17.98 ms and 32.54 ms. Meanwhile, the WHAM network system, which originally required a processing time of about 85.35 ms, was optimized to an average of 76.33 ms by reducing the computational accuracy through the implementation of the autocast technique on it, which improved the response speed of the system.

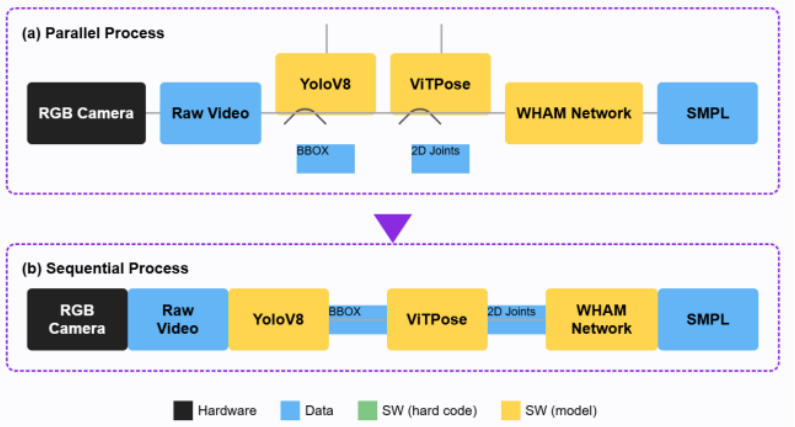

Fig 5. WHAM human body recognition process (a) Initial offline process (b) Improved online process

3. Process pipelining design: given that the interaction frequency with the hardware part of the robot is set to 50 Hz, and the current video recognition module requires a total of about 125 milliseconds, which is far less than the target frequency. Therefore, this study decided to use a pipelined approach for video processing optimization, as shown in Figure 6. The video motion capture module is partitioned into two sub-modules: the bounding box and feature extraction module (Yolo, ViTPose) and the SMPL recognition module (WHAM). The latter constitutes the critical path in the flow since it requires about twice the processing time of the former. With the pipeline optimization, the interaction cycle is reduced from the original about 125 ms to about 70 ms, which significantly improves the response performance of the system.

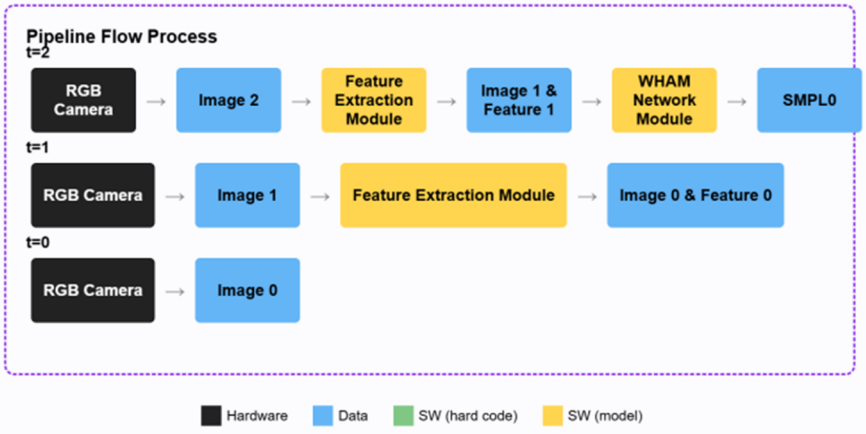

Fig 6. WHAM improved pipelined flow

4. Output frequency stabilization method: in order to solve the problem that the output frequency of the motion capture system still cannot reach 50hz and the output frequency is unstable, this study proposes the method of buffer pool combined with spherical interpolation algorithm. The method first establishes a message buffer queue for the SMPL identification data output from the WHAM system, with a queue length of 5. Each time the external output is made, the two most recent historical SMPL messages 150 milliseconds away from the current timestamp are selected according to the timestamps of the messages in the queue, and then the SLERP spherical interpolation algorithm is used to fuse these two SMPL messages to calculate the updated SMPL interpolation result and output it. This method effectively ensures that the output frequency is stabilized at 50 Hz to meet the requirements of the robot control system by forcing the WHAM system to set the output delay to 150 ms.

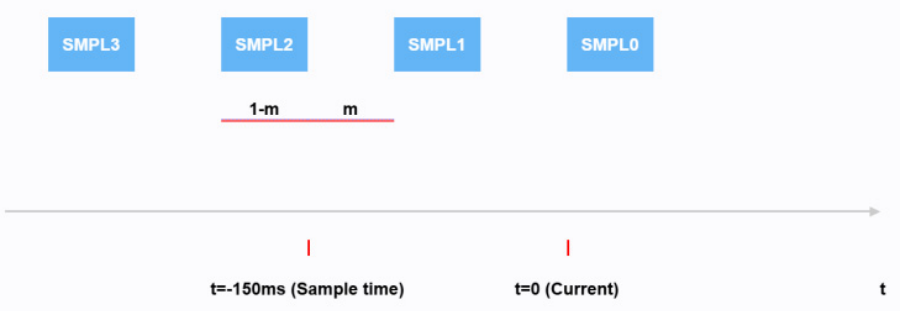

Fig 7. SMPL interpolation moment selection

The SLERP (Spherical Linear Interpolation) algorithm used in this study can be expressed as:

$$\mathrm{Slerp}(q_1, q_2, l) = \frac{\sin\big((1-m)\theta\big)}{\sin\theta}\cdot q_1 + \frac{\sin m\theta}{\sin\theta}\cdot q_2 \quad (1)$$

Where and denote the quaternion representation of the two key SMPL pose data, respectively, is the angle between the two quaternions, and is the interpolation parameter (value range [0,1]), which represents the relative position of the current time point between the two key frames.

3.3.2. Finger Movement Retargeting based on AR Devices

In this paper, Meta Quest 3-based ARKit Hand Tracking technology is used to capture the movements of the operator's fingers in real-time, and combined with Vuer and WebXR technologies to achieve efficient processing and visualization of the data. The system transmits the wrist position data to the upper limb retargeting algorithm through a bi-directional mechanism and accurately maps the finger movements to the robot through the Dex_retargeting algorithm to ensure natural and precise operation. At the same time, the system installs a high-resolution binocular camera on the robot's head, which transmits stereoscopic images back to the operator's AR device in real time to enhance immersion and operation precision. Through the integration of Meta Quest 3, Vuer, WebXR and Dex_retargeting, a complete closed-loop control system is constructed, which significantly improves the precision and efficiency of the robot's dexterous operation.

4. Experimental Results and Validation

4.1. Experimental Setup

The experimental equipment used in this paper is a host computer equipped with Intel i9-13900 CPU, Nvidia RTX3090Ti and 128G RAM. The monocular camera used for motion capture is the Greenlink CM678 camera, and the AR device used is Meta Quest3, and the python used in this paper is 3.8, the pinocchio version is 3.1.0, and the simulation environment is isaac gym.

4.2. Algorithm Accuracy Analysis

This paper first analyzes the algorithm accuracy. In this paper, the AMASS dataset is used as the experimental data, from which a total of 8145 trajectories are selected from the sub-datasets of ACCAD, CMU, JanpaneseEye, and KIT.

The evaluation metrics used in this paper for accuracy are (Mean Per Joint Position Error), which is the average distance error between the whole body of the robot and the corresponding node of the skeleton tree, and this item can be used to evaluate the ability of the redirection algorithm in tracking the position of the joints in millimeters. In addition, (Mean Per End-Effector Rotation Error) is designed to track the ability of the reorientation algorithm to track the rotational pose of the robot's end-effector (i.e., the end of the limbs) joints. The term uses the angular value of the axis angle of the pose error in radian production.

Table 3 below shows the experimental results of this paper:

Table 3. Comparison results of two algorithms

Method	$E_{mpjpe}(mm)\downarrow$	$E_{mpjpe\_elbow}(mm)\downarrow$	$E_{mpjpe\_wrist}(mm)\downarrow$	$E_{mpere}(rad)\downarrow$
Based on End-Effector Pose	27.744	53.322	2.167	0.066
Based on Skeleton Keypoint Pose	4.507	5.216	3.798	0.094

From Table 3, it can be seen that the skeleton tree key point pose-based method significantly outperforms the end-effector pose-based method in terms of joint position error. Specifically, in the upper limb part, the skeleton tree-based method only has an average error of 4.507 mm, which is reduced by about 83.8% compared to the 27.744 mm of the end-effector-based method; in the elbow joint part, the error of the skeleton tree-based method is 5.216 mm, which is reduced by about 90.2% compared to the 53.322 mm of the end-effector-based method. This result fully proves that the skeleton tree-based reorientation method proposed in this paper has significant advantages in maintaining the robot joint position accuracy.

4.3. Single AR Device Teleoperation Mode

In order to better demonstrate the effect of single AR device teleoperation mode, Figures 8 to 10 show the finger pose replication, dexterous grasping operation, and elbow-wrist mismatch with the realistic pose.

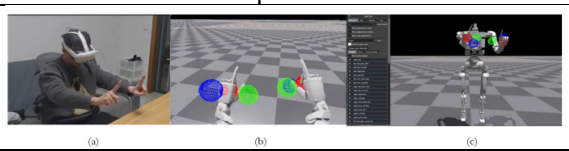

Fig 8. Finger gesture replication

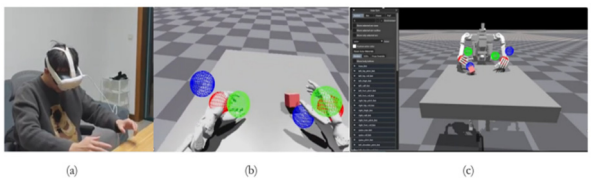

Fig 9. Dexterous grasping 1

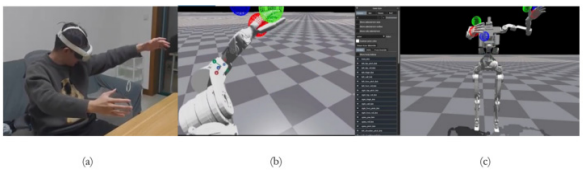

Fig 10. Elbow-wrist mismatch with realistic pose

4.4. Single WHAM Visual Capture Teleoperation Mode

To demonstrate the application of the single WHAM visual capture teleoperation mode, Figures 11 to 13 show the capture effects of the movements of the upper limb flat extension, hand waving and arm bending.

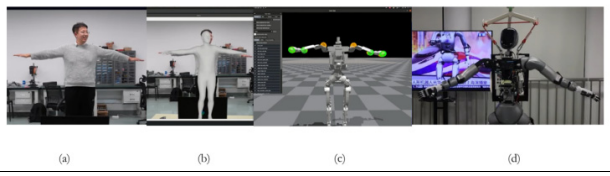

Fig 11. Upper Limb Extension

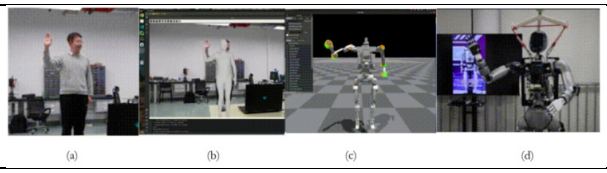

Fig 12. Hand waving

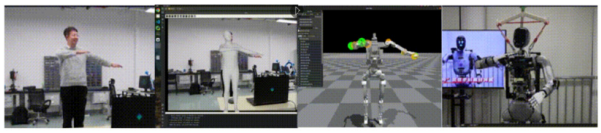

Fig 13. Bending arm

4.5. WHAM and AR Hybrid Teleoperation Mode

In order to demonstrate the advantages of the WHAM and AR hybrid teleoperation modes, Fig. 14 to Fig. 16 show the operation effects of reaching forward, reaching and grasping, and finger gesture replication.

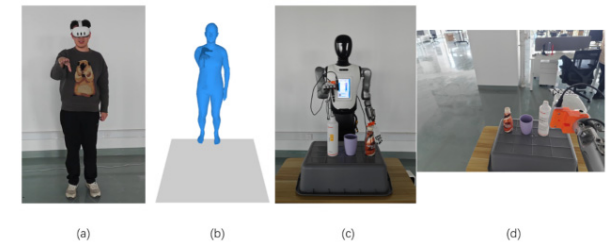

Fig 14. Reaching forward

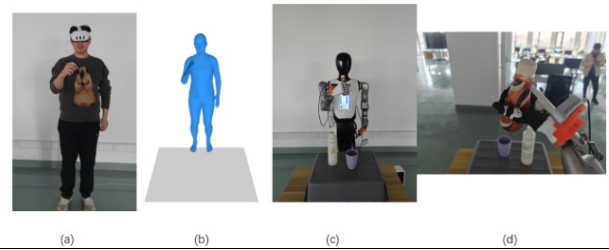

Fig 15. Reach and Grip

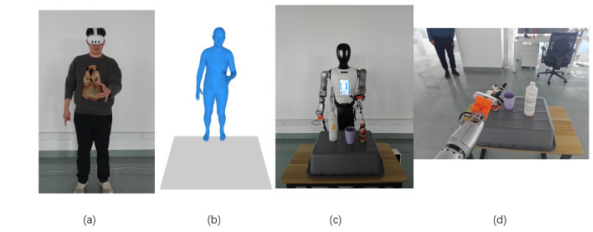

Fig 16. Finger gesture replication

5. Summary

In this study, a visual dexterous teleoperation method for robot upper limb based on human skeleton tree reorientation is proposed. Aiming at the problem that the existing visual teleoperation systems pay too much attention to the hand and neglect the overall attitude of the upper limb, this paper constructs a complete skeleton tree mapping framework, which realizes a high-precision conversion from the human body action to the robot motion.

The study first establishes a robot skeleton tree model, accurately locates the joint nodes and designs the logical parent-child relationship between the nodes. Subsequently, a two-stage mapping algorithm is proposed: the first stage maps the SMPL human model pose data to the robot skeleton tree, and the second stage solves the robot joint angles by an optimization algorithm. In order to achieve the real-time and accuracy of the system, two complementary motion capture methods are developed: upper limb motion capture based on WHAM and finger motion capture based on AR device. The WHAM algorithm is implemented by pipelined design and model lightweighting, while the buffer pool combined with spherical interpolation method is used to ensure the output frequency stability.

The experimental results show that the proposed hybrid teleoperation model fully combines the precise capture of upper limb wide range of movements by WHAM and the flexible control of finger fine movements by AR device, and shows significant advantages in accomplishing complex tasks, which not only ensures smooth and natural motion trajectories, but also provides high-precision control of the end-effector. This study provides a new research idea and realization way for human-robot collaboration in robotic upper limb teleoperation, which has important application value in the fields of hazardous environment operation, medical surgery and space exploration.

References

[1] Cheng X, Li J, Yang S, et al. Open-television: Teleoperation with immersive active visual feedback[J]. arXiv preprint arXiv:2407.01512, 2024.

[2] Shin S, Kim J, Halilaj E, et al. Wham: Reconstructing world-grounded humans with accurate 3d motion[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2024: 2070-2080.

[3] Qin Y, Yang W, Huang B, et al. Anyteleop: A general vision-based dexterous robot arm-hand teleoperation system[J]. arXiv preprint arXiv:2307.04577, 2023.

[4] Loper M, Mahmood N, Romero J, et al. SMPL: A skinned multi-person linear model[M]//Seminal Graphics Papers: Pushing the Boundaries, Volume 2. 2023: 851-866.

[5] Makoviychuk V, Wawrzyniak L, Guo Y, et al. Isaac gym: High performance gpu-based physics simulation for robot learning[J]. arXiv preprint arXiv:2108.10470, 2021.

[6] Ding R, Qin Y, Zhu J, et al. Bunny-visionpro: Real-time bimanual dexterous teleoperation for imitation learning[J]. arXiv preprint arXiv:2407.03162, 2024.

[7] Iyer A, Peng Z, Dai Y, et al. Open teach: A versatile teleoperation system for robotic manipulation[J]. arXiv preprint arXiv: 2403.07870, 2024.

[8] Mahmood N, Ghorbani N, Troje N F, et al. AMASS: Archive of motion capture as surface shapes[C]//Proceedings of the IEEE/ CVF international conference on computer vision. 2019: 5442-5451.