Supplementary Materials for

GuLiM: A Hybrid Motion Mapping Technique for Teleoperation of Medical Assistive Robot in Combating the COVID-19 Pandemic

Honghao Lv et al.

Corresponding author: Geng Yang, yanggeng@zju.edu.cn

The PDF file includes:

Fig. S1
Table S1
Instructions for source codes
ReadMe for video S1 and S2

Other Supplementary Materials for this manuscript includes the following:

Video S1 and S2
I. Specifications of The Used Medical Assistive Robot

There are four main parts of the self-designed teleoperated robot (Fig. 1): omnidirectional mobile chassis, dual-arm collaborative robot above the chassis, height adjustment mechanism, and other support devices. The chassis can move in all directions through different motion combinations of four Mecanum wheels, which is suitable for flexible movements in the narrow space of the isolation ward. The collaborative robot, YuMi, produced by ABB is chosen as the manipulator. An electric height adjustment mechanism is designed between the YuMi robot and the mobile chassis. In order to meet the confirmed needs of patient care, special replaceable connectors for various end effectors (Stylus Pen, Doppler ultrasound equipment, Handheld disinfection equipment, etc.) are designed. A storage box for carrying the medicine, disinfectants, and other equipment is installed on the side of the robot. A tablet computer is installed in the front of the robot, to conduct the remote daily medical checkups. Patients’ emotional state monitoring is also achieved based on the face data acquired using the camera on the tablet PC. The remote control is achieved via a pair of mini WiFi repeaters between the robot and the healthcare worker.

Table S1 lists general technical specifications of the medical assistive robot used in this work. As introduced in the paper, the YuMi robot is used to conduct the implementation of the teleoperated robot system. It is a dual-arm collaborative robot produced by ABB, and each arm has 7 Degrees of Freedom (DoF) and includes a smart gripper. The gripper has one basic servo module, communicating with the controller of YuMi robot over an Ethernet IP Fieldbus. The servo module is the basic part of the gripper. It gives the function of gripping objects. Fingers are installed on the base of the servo module, and finger movement and force can be controlled and supervised. The omnidirectional mobile chassis is driven by four servo motors, which are controlled by the corresponding servo motor controllers. The Controller Area Network (CAN-bus) communication mode is chosen between the mobile chassis servo controller and the master controller.
II. The Data Processing and Transmission Flow

As shown in Fig. S1, all IMUs are on straps which are worn on the body. Hub, the central processing unit of Perception Neuron, compiles and synchronizes motion data between IMUs and software through Wi-Fi. The NeuronDataReader SDK (provided by Noitom Technology Co., Ltd.) is used to receive and decode the motion data stream to the standard coordinate frame. The role of ROS in this work is to receive and process data from motion capture devices (including the motion data of upper limbs and hands), and to control the movement of the robotic arm and gripper. Two ROS nodes are set to receive and process the motion data of
upper limbs and hands respectively. The ROS node for upper limbs converts the upper-limb coordinate transform data from human to the pose data for the robot arm control. The ROS node for hand gestures processing converts the coordinate transform data of each finger to the quaternions of the fingertips for hand gesture recognition.

### III. Function Code for YuMi Robot Control Through ROS

In order to improve the repeatability of this work, and help the readers to reproduce the robot control, we open-sourced part of the function code for YuMi robot control in Github. If you don’t have a real YuMi robot, you also can do a simulation in the RobotStudio software using these codes.

**Requirements:**
- Operation System: Ubuntu16.04
- ROS version: Kinetic

**Source codes:**
1. ROS site code (python/C++):
   https://github.com/HonghaoLYU/YuMi-ROS_control_code_in_ROS
2. Robot site code/ RobotStudio simulation site (RAPID):
   https://github.com/HonghaoLYU/YuMi-ROS_control_sim_in_RobotStudio

### IV. README for Supplementary Videos

**Description:**
Two playable multimedia objects namely: GuLiM - Hybrid motion mapping (S1), Experiments & Verification (S2), are included as supplementary materials. These videos are to demonstrate more intuitively the proposed hybrid motion mapping solution (i.e., GuLiM). Video S1 shows a general process of the proposed hybrid mapping method. Video S2 shows the experiments setup and the comparative experiments with the Directly Mapping Method (DMM). The 1'05" to 1'15" of video S2 shows the challenging tasks and some application scenarios in the First Affiliated Hospital of Zhejiang University School of Medicine during the COVID-19 pandemic.

**Player Information:**
The video file is a standard mp4 format, which can be played on most media players including VideoLAN’s VLC Media Player (tested with version 3.0.11).

**Attachment filename:**
- Suppl_Video_S1_GuLiM - Hybrid motion mapping.mp4 (5.59 MB)
- Suppl_Video_S2_Experiments & Verification.mp4 (10.2 MB)

**Contact Information:**
Questions or comments regarding the video can be directed to Geng Yang at yenggeng@zju.edu.cn.