Cost-Effective Robot Arm Teleoperation via Human Pose Tracking with Monocular Camera

Bernard Parfaite[1], Luka Petrović[2], Vlaho-Josip Štironja[2], Ivan Marković[2], and Ivan Petrović[2]

[1] Universite Paul Sabatier Toulouse III, France
[2] University of Zagreb Faculty of Electrical Engineering and Computing, Croatia
parfaite.bernard@gmail.com

Abstract. Robot teleoperation is an essential tool for performing complex tasks that require dexterity beyond the capabilities of state-of-the-art algorithms. Existing teleoperation methods are often non-intuitive for human operators or require special sensors and equipment, making them cost-ineffective and impractical in many scenarios. In this paper, we propose a robot arm teleoperation framework that relies on monocular camera images. The proposed framework first uses lightweight neural networks for estimating the human operator body pose and recognizing their hand gesture. An efficient inverse kinematics algorithm then finds the desired robot arm configuration, achieving end-effector motion that imitates operator's wrist movement. Our teleoperation framework can be executed on an average laptop using a web camera with any robotic arm that has a ROS interface. We verified its performance in real-world experiments with a Kinova Jaco robotic arm, demonstrating the ability to grasp and move an object in the environment.

1 Introduction

In recent years, the development of robotic systems has witnessed significant advancements, revolutionizing various industries and sectors. One area of particular interest is robot arm teleoperation, where human operators can remotely control robotic arms to perform complex tasks in hazardous environments or situations requiring dexterity beyond the current capabilities of autonomous robots [1]. Existing teleoperation approaches [2][3][4] are often not intuitive and *natural* for human operators. Moreover, they are usually developed for a particular use case and may depend on specific characteristics of the teleoperated robot or its environment. Teleoperation methods that imitate the movement of the human operator with a robotic arm are more intuitive [5][6], but often require additional equipment which may be impractical. All of the mentioned properties limit usability of such teleoperation systems by increasing the deployment cost and requiring the operator to learn the control interface.

Some authors suggest using widely-available Android-based mobile devices as cheap teleoperation interfaces to increase cost-effectiveness [7][8]. Mobile devices

act as joystick controllers, which makes the teleoperation intuitive for the operator. However, it may be difficult for the operator to achieve complex robotic arm behavior, such as grasping an object and moving it to a different location. On the other hand, a cost-effective and intuitive teleoperation can be achieved by detecting the human full-body pose in RGB-D images and controlling the robot in a way that mimics the human operator [9][10][11]. These methods require depth sensing, which is usually not available on laptops and mobile devices, thus requiring an additional sensor.

To address these issues, in this paper we propose a cost-effective robot arm teleoperation framework that relies on monocular camera images. The proposed framework first detects the full-body pose of the human operator and their hand gesture, relying on lightweight open-source neural network models. An efficient inverse kinematics algorithm then finds the desired robot arm configuration that achieves end-effector motion which mimics the operator's wrist movement, enabling intuitive teleoperation. We make several deliberate choices to make the proposed framework as cost-effective as possible. First, we rely on a single monocular RGB camera, enabling teleoperation on virtually any laptop with a web camera or a mobile device. Second, for communication between all components in the system we utilize Robot Operating System (ROS), an open-source middleware framework that integrates control interface for many existing robots. Third, for human pose tracking and gesture recognition we use the open-source Mediapipe computer vision framework aimed at running in real-time on mobile devices and cheap hardware. The proposed teleoperation framework can thus be executed on an average laptop CPU with a web camera, using any robotic arm that has a ROS interface. We verified the performance of our framework in real-world experiments with the Kinova Jaco robotic arm, demonstrating the ability to grasp and move an object in the environment.

2 Robot arm teleoperation framework

To achieve cost-effective robot arm teleoperation, we first take as input a monocular camera image of the human operator. The obtained image is processed to estimate the full-body pose of the operator, and also to recognize hand gestures. From the full-body pose, we extract the information about wrist joint position of the person's arm. The relative movement of operator's wrist between consecutive frames is used to compute the robot control commands that mimic the person's arm movement, which are then sent to the robot for execution. The proposed teleoperation framework is shown in Fig. 1.

2.1 Human pose tracking

Human pose estimation using a monocular camera refers to the process of estimating positions of human body joints and limbs based solely on the input from a single camera, without explicitly tracking the movement of the person over time. This is an inherently ill-posed problem due to the need of estimating depth from

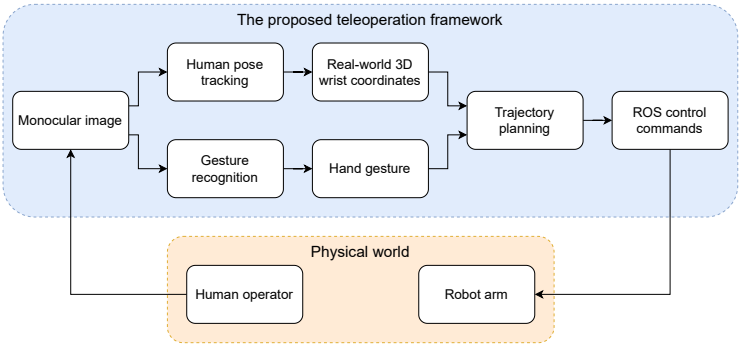


Fig. 1: The proposed teleoperation framework.

a single 2D image. State-of-the-art deep learning models are able to surmount this challenge, particularly if the intrinsic camera calibration parameters are known. In our framework, we employ the open-source pipeline called *Mediapipe*, a versatile platform designed to facilitate computer vision inferences over various sensory data, encompassing video and audio. Within this pipeline, we employ *MediaPipe Pose Landmark Detection*, a Google-developed open-source library based on deep learning that can accurately estimate 2D and 3D positions of human body keypoints (joints) in real-time, given a single monocular image. The implementation of Mediapipe enables high-fidelity body pose tracking, allowing the inference of 33 3D landmarks across the entire body from RGB video frames, as illustrated in Fig. 2.

The underlying deep model is a lightweight convolutional neural network BlazePose [12], tailored for real-time inference on mobile devices, allowing our teleoperation framework to work on low-power hardware. BlazePose first detects the human body from an image by employing an efficient face detector [13], under the assumption that a person's face is always visible for a single-person use case, such as the proposed teleoperation framework. Once a human body is detected, the model estimates positions of key body joints, often referred to as keypoints, including the head, shoulders, elbows, wrists, hips, knees, and ankles. Each keypoint is accompanied by a visibility score, signifying the likelihood of the keypoint being unobstructed and correctly positioned within the frame. BlazePose estimates the overall pose of the person by utilizing an encoder-decoder architecture to predict heatmaps for all joints, followed by another encoder that regresses directly to the coordinates of all joints [12]. The final output of the BlazePose human pose estimation system includes a set of 3D coordinates representing the positions of each joint.

We implement a ROS interface for Mediapipe in Python, where a monocular camera image is read from a ROS topic and the resulting human pose detection result can be published. After obtaining the human pose estimate, we extract the 3D position of the left hand (landmark 16 on Fig. 2), with the frame's origin positioned at the center between the hips. We denote this position with $\mathbf{p}_B^H$, in-

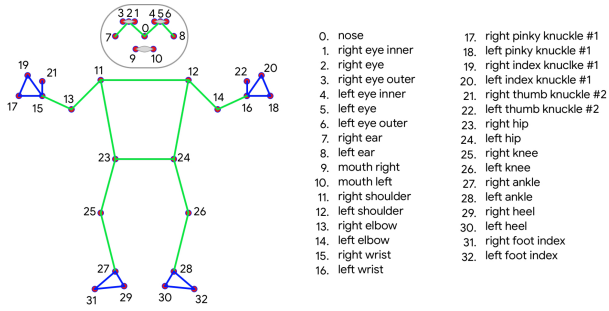


Fig. 2: The Mediapipe Pose Landmark Detection tracks 33 body landmarks that represent the approximate location of body parts [12].

dicating the hand position in the body frame. Our teleoperation framework only considers the relative change in the person's wrist position. Thus, our framework can operate even if the person is moving, while also circumventing the need for calibration between camera and robot arm frames, further enhancing simplicity and cost-effectiveness. Additionally, we utilize a hand gesture recognition classification model from the Mediapipe pipeline [14], enabling the detection of whether the human operator's hand is open or closed. The neural network model consists of 3 fully connected layers with 50 neurons each, acting as a lightweight real-time classifier.

2.2 Robot arm control

Once we acquire the real-world 3D coordinates of the person's left wrist in the body frame $\mathbf{p}_B^H$ and its corresponding hand gesture, we can plan the action to maneuver the robot arm's end effector to a desired location or control the gripper. We consider three high-level actions: teleoperation activation, robot arm movement, and gripper opening or closing. Before enabling arm movement teleoperation, we check that both wrists of the human operator are within the webcam's captured frame. Activation of teleoperation is achieved by the operator joining their hands.

To achieve the robot arm movement, we first compute the desired end-effector position in the robot base frame $\mathbf{p}_R^{EE}(k)$ for the k-th iteration by moving the arm proportionally to the detected wrist movement

$$\mathbf{p}_R^{EE}(t_k) = \mathbf{p}_R^{EE}(t_{k-1}) + \alpha\big(\mathbf{p}_B^H(t_k) - \mathbf{p}_B^H(t_{k-1})\big), \tag{1}$$

where α is the scaling factor. Then, we use Trac-IK [15] open-source library for efficient inverse kinematics to find the robot's configuration $\mathbf{q}(t_k)$ required to achieve the desired pose. We calculate the displacement between the desired and current configuration and compute a desired uniform joint space velocity that is passed to a PID velocity controller using ROS. Consequently, the robot arm's end effector mirrors the movement of the operator's hand.

The third action adjusts the state of the gripper based on *palm open* and *palm closed* gestures. Based on the detected gestures, we wither fully open or fully close the gripper fingers. The set of actions is modular and different actions can be derived from hand gestures and used for different robot arms depending on specific robot features. Our framework is applicable to all robot arms that have a ROS interface. Moreover, given that the pose tracking can be performed for both wrists, it is feasible to teleoperate two arms simultaneously using a single monocular camera.

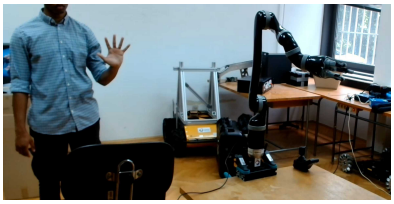
3 Results

To evaluate the proposed teleoperation framework, we conducted a performance analysis in simulation and real-world demonstrations in a laboratory environment on the 6 degree-of-freedom Kinova Jaco robot arm. First we examined the conditions that enable teleoperation with a cheap web camera in a number of experiments with a simulated robot arm. For simulation, we used Gazebo and the Kinova Jaco robot arm URDF model, which exhibits the same behavior as the real-world arm due to the shared ROS interface. Then, we execute three real-world teleoperation scenarios that include grasping and moving a plastic bottle.

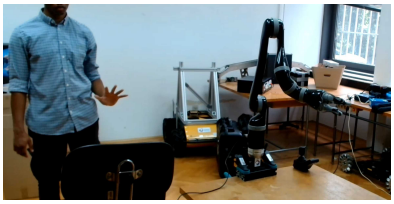
3.1 Simulation performance analysis

First, we investigated the human pose tracking accuracy given different light conditions and human posture. The precision of pose tracking is notably improved when the human operator is situated in well-illuminated surroundings and when the upper body is fully captured within the camera frame. Consequently, we choose to compute control commands only if the left wrist keypoint has visibility greater than 0.9. Next, we observed enhanced accuracy in hand gesture recognition when the palm faces the camera and maintains a distance of no more than 2 m, coupled with favourable lighting conditions. Note that the BlazePose model in poor light conditions sometimes erroneously classifies the *palm closed* gesture as the *thumbs up* gesture, leading to a missed gripper action.

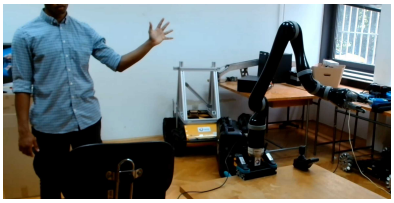
We then investigated the robot arm motion, which was smooth and accurately tracked the calculated trajectory, in most cases achieving the intended movement commanded by the user's hand gestures. However, we found that the BlazePose depth estimation often lacked accuracy, which was expected due to the ill-posed problem of monocular depth estimation, especially using a cheap camera. To address this limitation and achieve teleoperation accurate enough for completing complex movements, such as grasping a bottle, we decided to restrict the movement of the end effector to a pre-specified distance of ± 0.1 m along the axis corresponding to the camera depth, if any change in the operator's arm movement along the camera depth axis is observed. In summary, given good lighting conditions and accurate human operator pose estimation, our simulation performance study indicates the feasibility of accomplishing pick and place tasks using the developed teleoperation framework.



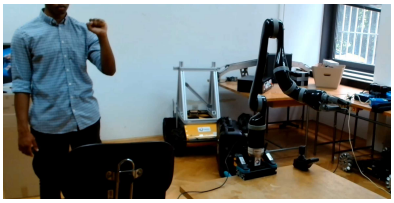
(a) Starting position



(b) Move down



(c) Move left

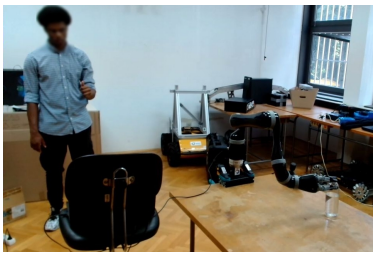


(d) Close fist

Fig. 3: Teleoperation in free space.



(a) Starting position



(b) Grasp the bottle



(c) Move the bottle

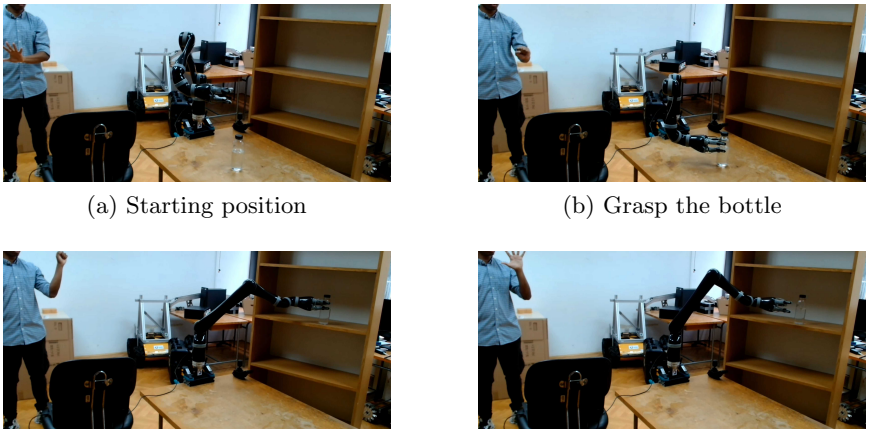


(d) Release the bottle

Fig. 4: Manipulating the plastic bottle.

3.2 Real-world demonstrations

In order to test our teleoperation framework, three scenarios have been considered. In the first scenario, we verify that our framework enables the motion of the arm in free space that mimics the movement of the human operator. Moreover,



(a) Starting position (b) Grasp the bottle

(c) Move to shelf (d) Release the bottle

Fig. 5: Moving the plastic bottle from the table to the shelf.

we verify that the robot arm gripper closes in response to the operator's palm closure, and that it opens when the palm is extended, as shown in Fig. 3. The latency between operator's hand movement of about 0.3 m and robot's motion completion was around 1.2 s. Note that the robotic arm's speed was deliberately restricted during our testing phase for safety considerations.

In the second scenario, the goal is to teleoperate the robot to move an empty plastic bottle. The operator guides the robot to grasp the bottle from a table, move it in the air, and then return it on the same table without flipping, as shown in Fig. 4. We completed this process 5 times in a row to ensure repeatability.

The third scenario considered moving the plastic bottle from a table to a shelf. The operator guided the robotic arm towards the bottle on the table, grasped it using hand gestures, moved it towards the shelf and successfully released it, as depicted in Fig. 5.

4 Conclusion

In this paper, we developed a framework that enables intuitive robot arm teleoperation using a monocular camera. The proposed framework first uses lightweight neural networks for estimating the human operator body pose and recognizing their hand gesture. An efficient inverse kinematics algorithm is then utilized to find the robot arm configuration, achieving end-effector motion that mimics operator's wrist movement. The proposed teleoperation framework can be executed on an average laptop with any robotic arm that has a ROS interface. We verified the performance of our teleoperation framework in real-world experiments with a Kinova Jaco robotic arm, demonstrating the ability to grasp and move an object in the environment.

Acknowledgment. This research has been supported by the European Regional Development Fund under the grant PK.1.1.02.0008 (DATACROSS).

References

1. Vertut, J.: Teleoperation and robotics: applications and technology, vol. 3. Springer Science & Business Media (2013)
2. Kofman, J., Wu, X., Luu, T.J., Verma, S.: Teleoperation of a robot manipulator using a vision-based human-robot interface. IEEE tran. on industrial electronics 52(5), 1206–1219 (2005)
3. Hainsworth, D.W.: Teleoperation user interfaces for mining robotics. Autonomous Robots 11(1), 19–28 (2001)
4. Birk, A., Schwertfeger, S., Pathak, K.: A networking framework for teleoperation in safety, security, and rescue robotics. IEEE Wireless Communications 16(1), 6–13 (2009)
5. Artemiadis, P.K., Kyriakopoulos, K.J.: Emg-based control of a robot arm using low-dimensional embeddings. IEEE transactions on robotics 26(2), 393–398 (2010)
6. Gupta, G.S., Mukhopadhyay, S.C., Messom, C.H., Demidenko, S.N.: Master–slave control of a teleoperated anthropomorphic robotic arm with gripping force sensing. IEEE Transactions on Instrumentation and Measurement 55(6), 2136–2145 (2006)
7. Szymańska, E., Petrović, L., Marković, I., Petrović, I.: Mobile robot teleoperation via android mobile device with udp communication. In: 2021 44th International Convention on Information, Communication and Electronic Technology (MIPRO). pp. 1143–1148. IEEE (2021)
8. Yepes, J.C., Yepes, J.J., Martinez, J.R., Pérez, V.Z.: Implementation of an android based teleoperation application for controlling a kuka-kr6 robot by using sensor fusion. In: Pan American Health Care Exchanges (PAHCE). pp. 1–5. IEEE (2013)
9. Syakir, M., Ningrum, E.S., Sulistijono, I.A.: Teleoperation robot arm using depth sensor. In: International Electronics Symposium (IES). pp. 394–399. IEEE (2019)
10. Marić, F., Jurin, I., Marković, I., Kalafatić, Z., Petrović, I.: Robot arm teleoperation via rgbd sensor palm tracking. In: 2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO). pp. 1093–1098. IEEE (2016)
11. Handa, A., Van Wyk, K., Yang, W., Liang, J., Chao, Y.W., Wan, Q., Birchfield, S., Ratliff, N., Fox, D.: Dexpilot: Vision-based teleoperation of dexterous robotic hand-arm system. In: 2020 IEEE International Conference on Robotics and Automation (ICRA). pp. 9164–9170. IEEE (2020)
12. Bazarevsky, V., Grishchenko, I., Raveendran, K., Zhu, T., Zhang, F., Grundmann, M.: Blazepose: On-device real-time body pose tracking. arXiv preprint arXiv:2006.10204 (2020)
13. Bazarevsky, V., Kartynnik, Y., Vakunov, A., Raveendran, K., Grundmann, M.: Blazeface: Sub-millisecond neural face detection on mobile gpus. arXiv preprint arXiv:1907.05047 (2019)
14. Sung, G., Sokal, K., Uboweja, E., Bazarevsky, V., Baccash, J., Bazavan, E.G., Chang, C.L., Grundmann, M.: On-device real-time hand gesture recognition. arXiv preprint arXiv:2111.00038 (2021)
15. Beeson, P., Ames, B.: Trac-ik: An open-source library for improved solving of generic inverse kinematics. In: 2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids). pp. 928–935. IEEE (2015)