

# MARVIN: Remote Teleoperation of a Dual-Arm Upper-Body Avatar Robot

Real-time human motion mirroring with client-side ML and ROS2

Jaeho Cho\*

jaeho.cho@cooper.edu

The Cooper Union for the Advancement of Science and Art  
New York, New York, USA

Sophia Klymchuk\*

sophia.klymchuk@cooper.edu

The Cooper Union for the Advancement of Science and Art  
New York, New York, USA

## ABSTRACT

We present MARVIN, a dual-arm teleoperational robot that mirrors a human operator's upper-body motion in real time using just a standard webcam. Client-side MediaPipe models extract 3D pose and hand landmarks, which are transmitted via websocket to a ROS2-MoveIt servoing stack that commands two OpenManipulator-X arms. We detail perception-to-actuation mappings, including geometric formulations for shoulder flexion, shoulder abduction/adduction, and elbow flexion, and describe a web interface that enables intuitive interaction. We report latency, accuracy, and user study results from local and remote operation, discuss safety and failure modes, and outline pathways toward full-body telepresence.

## CCS CONCEPTS

- Hardware;

## KEYWORDS

teleoperation, human-robot interaction, ROS2, MoveIt, MediaPipe

### ACM Reference Format:

Jaeho Cho and Sophia Klymchuk. 2026. MARVIN: Remote Teleoperation of a Dual-Arm Upper-Body Avatar Robot: Real-time human motion mirroring with client-side ML and ROS2. In *Proceedings of TEI Student Design Challenge (TEI SDC '26)*. ACM, New York, NY, USA, 3 pages. <https://doi.org/XXXXXXX>

## 1 INTRODUCTION

MARVIN is a teleoperational robot that mirrors the upper-body movements of a human operator in real time. Operation works via a local webcam or a remote connection. For remote operation, a web application captures the user's webcam stream for pose landmarking, enabling MARVIN to act as a physical avatar across any distance.

## 2 SDC 2026 THEME: SENSORY RITUALS

MARVIN explores the theme of Sensory Rituals through the design of a remotely operated robot that reintroduces physicality into digital communication. In contemporary society, much of interpersonal interaction occurs through screen-based interfaces, which often diminish embodied and sensory engagement. The proposed system seeks to restore a sense of presence and tactility within virtual meetings by allowing users to inhabit a physical robot equipped with two controllable robotic arms.

The act of remotely manipulating the robot becomes a new form of ritualized sensory engagement—practice that bridges physical and digital domains to cultivate social connection, attentional focus, and emotional resonance. By translating human gestures into robotic motion, the system enables users to perform actions, express intention, and interact with shared physical environments. This transforms otherwise disembodied virtual communication into an embodied, tangible experience.

Furthermore, the robot's accessibility—requiring only a standard webcam and web interface—extends this ritual of presence to anyone, anywhere, without technical barriers. This inclusivity positions the system as a tool for fostering shared sensory experience and collective grounding in hybrid or remote contexts. Through its integration of tangible interaction, telepresence, and accessible design, the project proposes that sensory rituals need not be confined to traditional or personal practices; they can also emerge from technological mediation that reconnects individuals to the physical world and to one another.

## 3 HARDWARE

MARVIN uses components from the ROBOTIS OpenManipulator-X platform. Each arm has 5 DOF and is powered by Dynamixel XM430-W350 servos. We employ two arms attached to an aluminum-beam torso and powered via a Dynamixel U2D2 power hub.

## 4 SOFTWARE

MARVIN runs on the Humble distribution of ROS2. High-level motion is handled by MoveIt, which translates user commands into low-level commands sent to Dynamixel controllers. A custom URDF/Xacro model encodes simultaneous kinematics, inertias, and control interfaces for both arms.

### 4.1 Pose Detection

OpenCV interfaces a MediaPipe-based pose detector to extract normalized 3D joint landmarks from a webcam stream [1]. We use

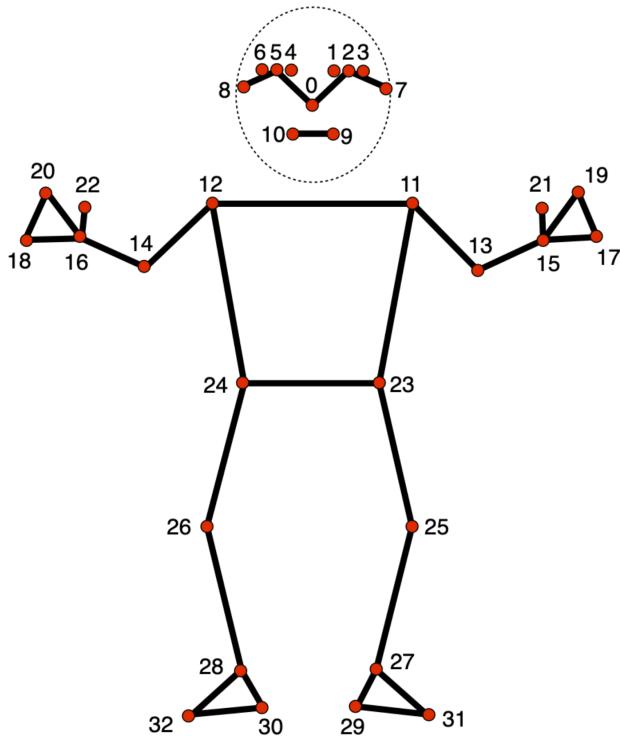
\*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

TEI SDC '26, March, 2026, Chicago, Illinois

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM.  
ACM ISBN 978-1-4503-XXXX-X/2026/06  
<https://doi.org/XXXXXXX.XXXXXXX>

shoulder (11,12), elbow (13,14), wrist (15,16), and hip (23,24) landmarks to compute joint angles that are then mapped to MARVIN's kinematic model.



**Figure 1: MediaPipe Pose Landmarks.** The pose landmark model tracks 33 body landmark locations, representing the approximate location of the labeled body parts.

**4.1.1 Shoulder Flexion.** Let  $S$  (shoulder),  $E$  (elbow),  $H$  (same-side hip),  $H_{opp}$  (opposite hip).

$$\mathbf{u} = E - S, \quad \mathbf{v} = H - S, \quad \mathbf{h} = H_{opp} - H, \quad \hat{\mathbf{n}} = \frac{\mathbf{h}}{\|\mathbf{h}\|}$$

Project onto the midline plane orthogonal to  $\hat{\mathbf{n}}$ :

$$\mathbf{u}_\pi = \mathbf{u} - (\mathbf{u} \cdot \hat{\mathbf{n}}) \hat{\mathbf{n}}, \quad \mathbf{v}_\pi = \mathbf{v} - (\mathbf{v} \cdot \hat{\mathbf{n}}) \hat{\mathbf{n}}.$$

Flexion:

$$\alpha = \arccos \left( \frac{\mathbf{u}_\pi \cdot \mathbf{v}_\pi}{\|\mathbf{u}_\pi\| \|\mathbf{v}_\pi\|} \right). \quad (1)$$

Map  $\alpha$  to shoulder flexion/extension joint ( $J_1$ ).

**4.1.2 Shoulder Adduction/Abduction.** Let  $\mathbf{u} = E - S$ ,  $\hat{\mathbf{n}}$  as above. Project  $\mathbf{u}$ :

$$\mathbf{u}_\pi = \mathbf{u} - (\mathbf{u} \cdot \hat{\mathbf{n}}) \hat{\mathbf{n}}.$$

Adduction magnitude:

$$\beta = \arccos \left( \frac{\mathbf{u} \cdot \mathbf{u}_\pi}{\|\mathbf{u}\| \|\mathbf{u}_\pi\|} \right). \quad (2)$$

Signed ab/adduction can be obtained from  $\text{sign}((\mathbf{u} \times \mathbf{u}_\pi) \cdot \hat{\mathbf{n}})$ . Map to  $J_2$ .

**4.1.3 Elbow Flexion.** Let  $\mathbf{u} = E - S$  and forearm  $\mathbf{f} = E - W$ . Then

$$\theta = \arccos \left( \frac{\mathbf{u} \cdot \mathbf{f}}{\|\mathbf{u}\| \|\mathbf{f}\|} \right). \quad (3)$$

Map  $\theta$  to elbow joint  $J_3$ .

**4.1.4 Hand Open/Close.** Using the Hand Landmarker, define a reference length from wrist (0) to middle-finger MCP (9). Mark a hand as open if fewer than three fingertips among indices  $\{4, 8, 12, 16, 20\}$  lie closer to the wrist than the reference. Publish binary open/closed to drive the gripper.



**Figure 2: MediaPipe Hand Landmarks.** The hand landmark model bundle detects the keypoint localization of 21 hand-knuckle coordinates within the detected hand regions.

## 4.2 Control

We use MoveIt real-time servoing with two independent kinematic chains, one per arm. Each chain is associated with a servoing node that subscribes to desired joint velocities. Desired velocities are computed from the error between detected target joint angles and current joint states from the `/joint_states` topic. Errors are scaled by gains and sent as JointJog commands.

## 5 WEBSITE

The MediaPipe Pose & Hands – ROSBridge webpage functions as the browser-based front end of MARVIN's remote teleoperation system. It captures the user's webcam stream, performs on-device pose and hand landmark inference, and transmits the resulting kinematic data to the ROS2 backend in real time. The interface also embeds a live video stream of the robot, allowing visual feedback during operation. Implemented entirely in client-side JavaScript using the MediaPipe Tasks API, the system requires no native installation or external dependencies. A compact control panel enables users to start or stop the camera, select inference model complexity, and toggle streaming to the ROSBridge server, which exposes `pose_landmarks` and `hand_landmarks` topics for downstream motion control.

Detected landmarks are overlaid directly on the live video feed, and the interface reports both frame rate and end-to-end latency to the ROSBridge server, enabling quantifiable performance evaluation under varying network conditions. Hand openness is computed geometrically from landmark distances, and both hand-state and pose data are serialized into ROS-compatible JSON messages. This design supports seamless integration with MARVIN's MoveIt servoing pipeline, where landmark-derived joint angles are mapped to actuator commands for real-time mirroring of human motion.

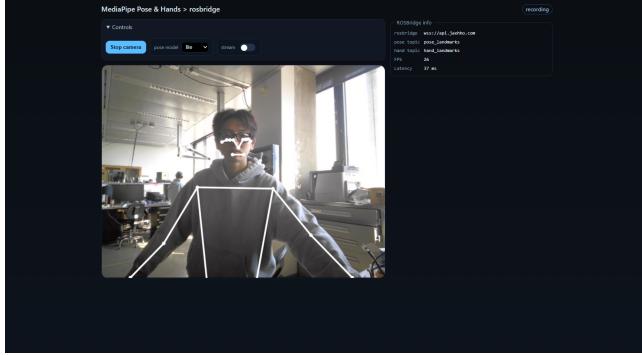


Figure 3: Browser UI for teleoperation. TODO: elaborate.

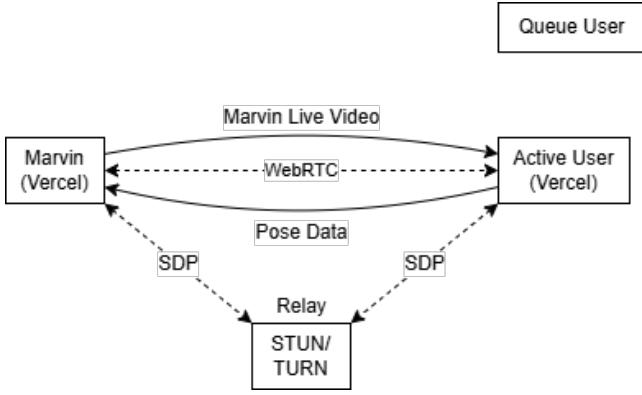


Figure 4: ROS rqt graph overview. TODO: elaborate.

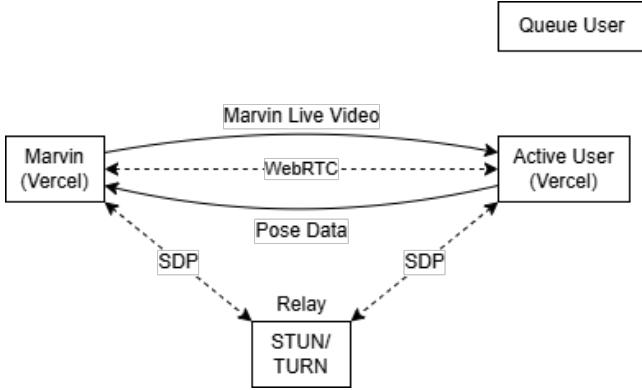


Figure 5: Communication pipeline and timing. TODO: elaborate.

## 6 DISCUSSION

MARVIN demonstrates that fully client-side inference is viable for dual-arm mirroring. Remaining gaps include wrist pronation/supination estimation, depth ambiguity in monocular input, and gripper force control. Future work includes fusing depth, adding self-calibration, and extending to mobile bases for whole-body telepresence.



Figure 6: MARVIN hardware. Front and oblique views of the dual-arm upper-body platform with aluminum-beam torso, two 5-DOF OpenManipulator-X arms, and parallel grippers. Insets show Dynamixel XM430-W350 servos, U2D2 hub, and mounting.

## 7 ETHICAL AND SOCIETAL IMPACT

Telepresence expands access but raises safety and privacy concerns. We log minimal data, anonymize telemetry where possible, and provide hard e-stop. Future deployments must consider operator authentication and bystander consent in public spaces.

## 8 CONCLUSION

We provided a browser-first teleoperation system that minimizes setup while retaining precise, low-latency control of a dual-arm avatar robot. We plan to release code and evaluation datasets.

## ACKNOWLEDGMENTS

We thank the Cooper Union community for volunteering and supporting facilities.

## REFERENCES

- [1] [n. d.]. *MediaPipe Solutions guide / Google AI Edge*. <https://ai.google.dev/edge/mmediapipe/solutions/guide>

## A ANGLE MAPPING AND CALIBRATION

We recommend a short calibration where the operator performs canonical poses to establish shoulder plane and joint-neutral offsets. Offsets are subtracted from observed angles before mapping to robot joints.

## B IMPLEMENTATION DETAILS

Browser stack: TypeScript, WebAssembly MediaPipe Tasks. ROS-Bridge schema and message formats are in the repository. Build and launch instructions for both simulation (Gazebo/Ignition) and hardware are provided.

Received NA; revised NA; accepted NA