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

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

JAEHO CHO*, The Cooper Union for the Advancement of Science and Art, USA
SOPHIA KLYMCHUK*, The Cooper Union for the Advancement of Science and Art, USA

MARVIN is 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**;

Additional Key Words and Phrases: 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, 7 pages. https://doi.org/XXXXXXX.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, where a web application captures the user's webcam stream for pose landmarking, enabling MARVIN to act as a physical avatar across any distance.

MARVIN explores the theme of the 2026 SDC theme of Sensory Rituals through the design of a remotely-operated humanoid robot that reintroduces physicality into digital communication. Contemporarily, much of interpersonal digital interaction occurs via screen-based interfaces that flatten embodied and sensory engagement. Via MARVIN, we seek to restore a sense of physical presence to virtual meetings by allowing its users to inhabit a robot equipped with two controllable arms.

By remotely manipulating the robot via one's own mirrored actions, tangible social connection can be had at-distance, enabling the operator to perform actions, express intentions, and interact with physical environments shared by the receiver.

*Both authors contributed equally to this research.

Authors' addresses: 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.

Furthermore, MARVIN is an accessible platform, as it requires only a standard webcam and Internet interface, which extends this ritual to anyone irrespective of distance, with only minimal technological barriers. This inclusivity broadens the system's use as a tool for fostering shared sensory experiences in remote or hybrid environments. Via its integration of presence, interaction, and accessibility, MARVIN allows tangible rituals to emerge out of traditional confinement and into a process of technological mediation that reconnects individuals to the physical world and to one another.

2 HARDWARE

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

3 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, inertial, and control interfaces for both arms.

3.1 Pose Detection

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

3.1.1 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:

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

Map α to shoulder flexion/extension joint (J_1).

3.1.2 Shoulder Adduction/Abduction. Let $\mathbf{u}$, $\hat{\mathbf{n}}$ be as above. Project $\mathbf{u}$: $\mathbf{u}_\pi = \mathbf{u} - (\mathbf{u} \cdot \hat{\mathbf{n}})\,\hat{\mathbf{n}}$.
Adduction magnitude:

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

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

3.1.3 Elbow Flexion. Let $\mathbf{u}$ be as above and forearm $\mathbf{f} = E - W$. Then

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

Map θ to elbow joint J_3.

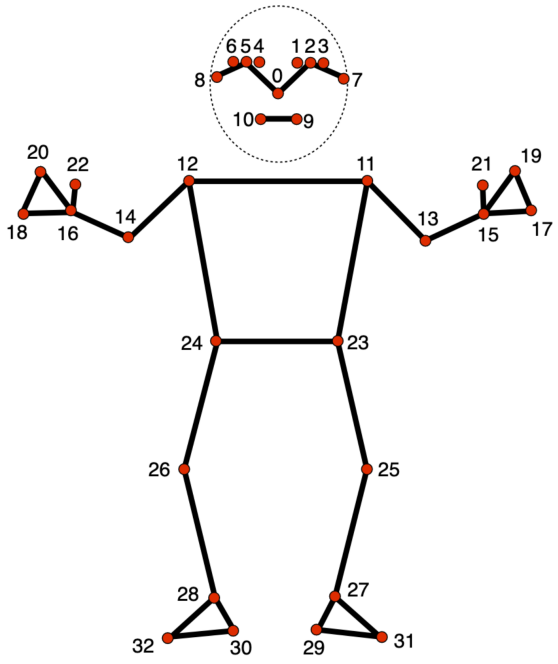

Fig. 1. MediaPipe Pose Landmarks. The pose landmarker model tracks 33 body landmark locations, representing the approximate location of the labeled body parts.

3.1.4 Hand Open/Close. Using the Hand Landmarker (labels shown in Figure 2), define a reference length from wrist (0) to middle-finger MCP (9). We 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. We then publish binary open/closed messages to drive the gripper.

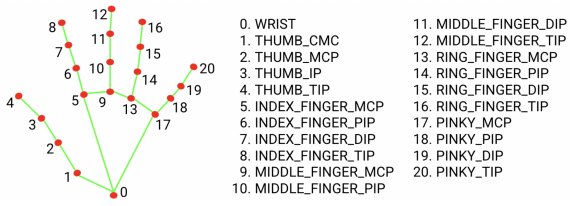

Fig. 2. MediaPipe Hand Landmarks. The hand landmark model bundle detects the keypoint localization of 21 hand-knuckle coordinates within the detected hand regions.

3.2 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. These velocities are computed from the error between detected target joint angles and current joint states from the `/joint_states` topic, before being scaled by gains and sent as `JointJog` commands.

4 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.

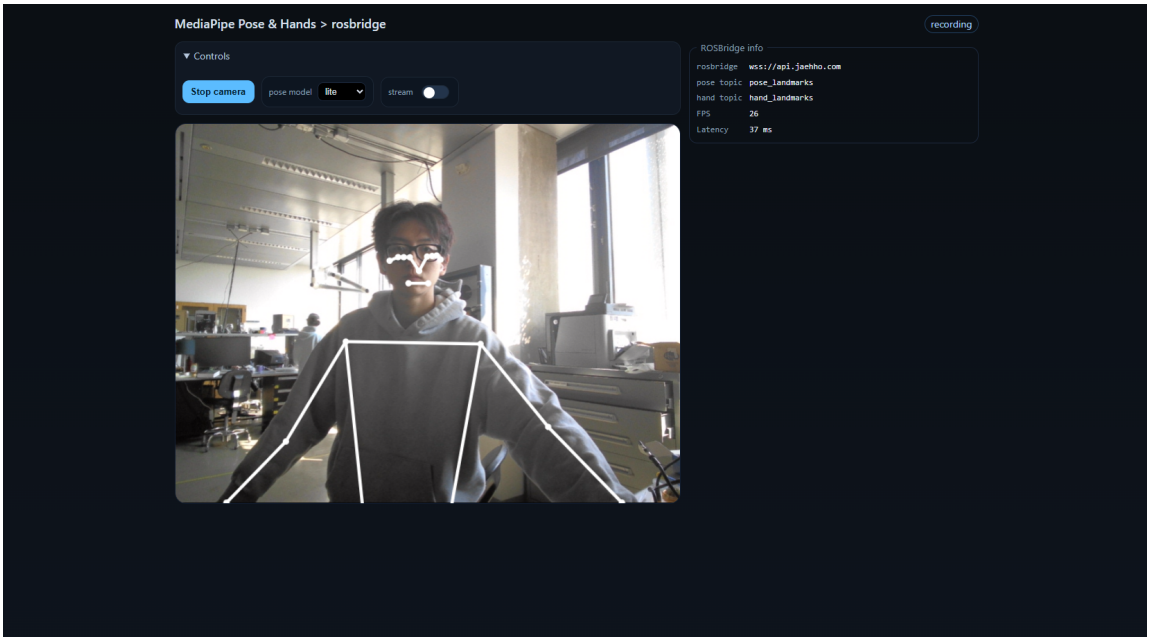

Fig. 3. Browser UI for teleoperation. TODO: elaborate.

5 OPERATION AND USER STUDY

MARVIN was tested in local operation during the Cooper Union End of Year Show of May 2025. The robot mirrored onlooker movements in real time, and we received qualitative feedback requesting the addition of hand gestures for a more natural feeling. This informed subsequent implementation of the hand landmark module and gripper control.

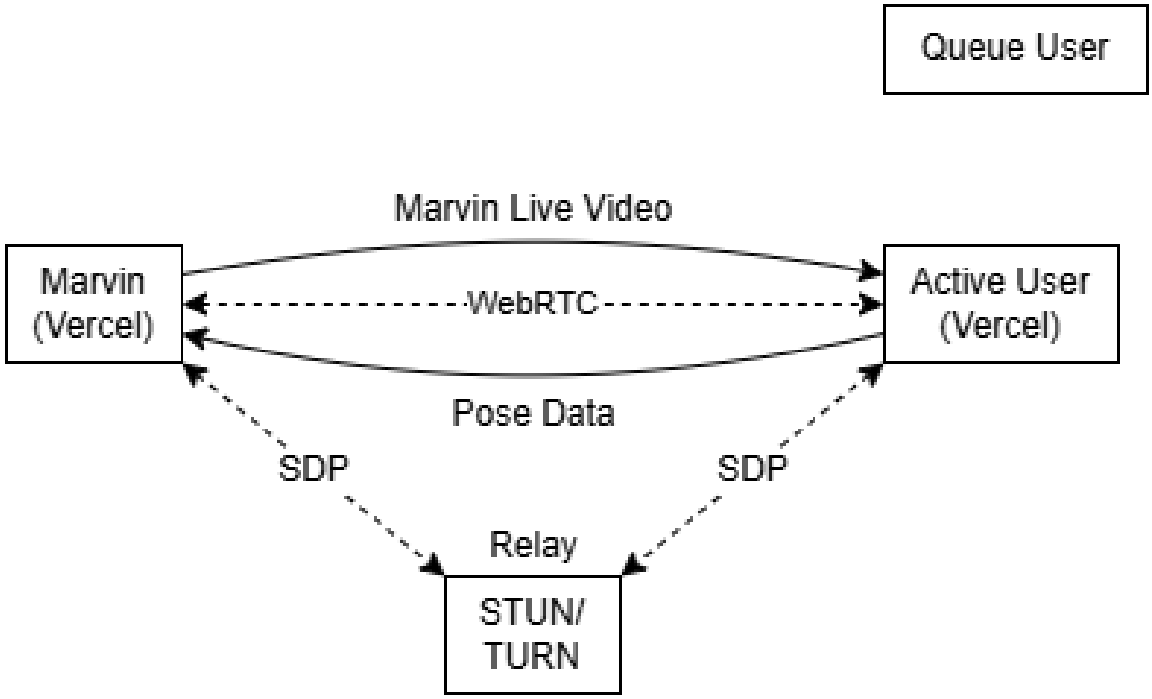

Fig. 4. ROS rqt graph overview. TODO: elaborate.

We additionally tested remote operation of MARVIN with peer volunteers from the Cooper Union community, achieving low latency? blah blah.

5.1 Latency

We define end-to-end latency as camera exposure to actuator motion onset. We report median and 95th percentile latencies for both conditions, along with landmark inference time and ROSBridge transmission time.

Table 1. Performance summary (illustrative; replace with measured data).

Metric	Local	Remote
End-to-end latency (ms)	*TODO*	*TODO*
Pose inference (ms/frame)	*TODO*	*TODO*
FPS (avg)	*TODO*	*TODO*
Task success rate (%)	*TODO*	*TODO*
NASA-TLX (0–100)	*TODO*	*TODO*

6 SAFETY AND LIMITS

We enforce velocity, position, and torque limits at the controller. We apply workspace clamping and filtering of pose outliers using temporal smoothing. Failure modes include landmark jitter, partial occlusion, and network loss. A dead-man switch disables actuation when no fresh landmarks arrive within a timeout.

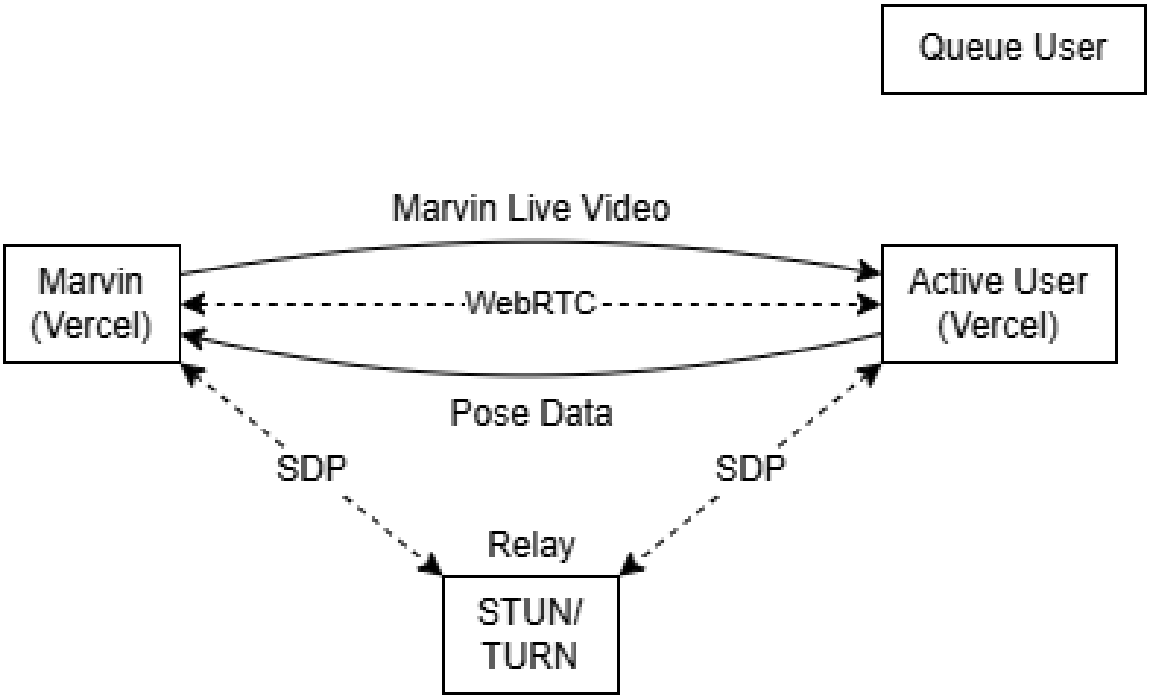

Fig. 5. Communication pipeline and timing. TODO: elaborate.

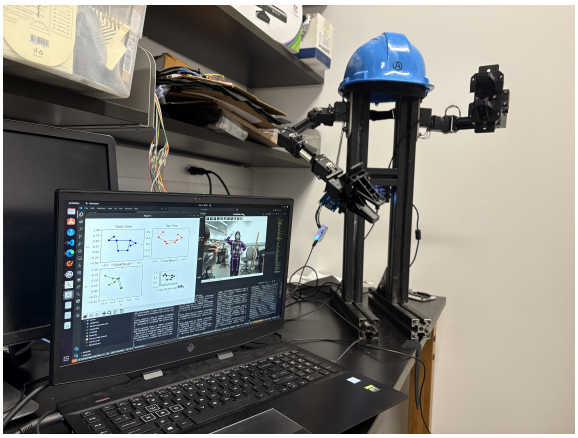

Fig. 6. MARVIN hardware. Front and oblique views of the dual-arm upper-body platform with aluminum-beam torso, two 5-DOF OpenManipulator-X arms, and parallel grippers. Insets show Dynamixel XM430-W350 servos, U2D2 hub, and mounting.

7 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.

8 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.

9 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/mediapipe/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. ROSBridge 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