Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Surgical Intelligence Labs: Collecting First-Person Surgical Data with a Smartphone to Train Autonomous Robots

Mustafa Khan

mustafa.t.khan@uth.tmc.edu

The University of Texas Health Science Center at Houston

João Bombardelli

The University of Texas Health Science Center at Houston

Research Article

Keywords: Egocentric video, Operating room, Foundation model, Hand tracking, Robotic surgery, Smartphone

Posted Date: November 13th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11654389/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Surgical Intelligence Labs: Collecting First-Person Surgical Data with a Smartphone to Train Autonomous Robots

Mustafa Khan, DO[1] · João Bombardelli, MD[1]

[1] Division of Plastic Surgery, UTHealth Houston, Houston, Texas, USA

Author information

Mustafa Khan: ORCID 0000-0002-5430-7572

João Bombardelli: ORCID 0000-0002-2293-7967

Corresponding author

Mustafa Khan, DO
Division of Plastic Surgery, UTHealth Houston
Email: mustafa.t.khan@uth.tmc.edu

Statements and Declarations

Funding. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Competing interests. The authors declare no relevant financial or non-financial competing interests.

Author contributions. All authors contributed equally to all parts of this work.

Ethics approval. No patients or study participants were recorded in this study. The pilot recordings were made by the authors in a non-clinical setting. Institutional review board approval was therefore not required.

Consent to participate. Not applicable. No patients or study participants were recorded.

Consent to publish. Not applicable.

Data availability. The data supporting this study are available from the corresponding author on reasonable request.

Use of generative AI. Generative artificial intelligence tools were used to assist with coding, debugging and validation of the application, and to assist with the grammar and clarity of the manuscript. All outputs were reviewed, approved and overseen by the authors, who take full responsibility for the content of the manuscript.

Abstract

Background. Robot foundation models learn from large, diverse data, and human video is becoming a powerful way to supply it. Surgery has almost none beyond console video, yet the hands of surgeons, assistants and scrub technologists, seen from their own eyes, could be the main source to train autonomous robots in the operating room (OR).

Methods. We built Surgical Intelligence Labs, a smartphone app that turns a clinician's iPhone into a first-person (egocentric) data-collection device. A head-mounted phone records tasks from operative field and back-table setup to console and microscope views, saving each frame with its 6-degree-of-freedom (6-DoF) head pose from Apple's ARKit and 50 Hz device motion. Hands are tracked on the phone (21 key points each), lifted to three dimensions by fitting an articulated hand model, and retargeted to a Unitree G1+ humanoid with five-finger Wuji Hand 2 hands and to ASIA, our bimanual scrub-technician robot. OR recording requires institutional approval, consent and on-device de-identification.

Results. On a 63-s suturing exercise recorded with a head-mounted iPhone 17, ARKit tracked the head in all 1,877 video frames, with the phone a mean 55° below the horizon, inside the framing guide's target band. Both hands were tracked on the phone in 97% of sampled frames, and the 3D hand fit was kept for 99% of hand observations. Position-based retargeting to five-finger Wuji Hand 2 hands kept thumb-to-fingertip gaps in pinches and grips within 2 mm of the fitted human gaps (medians). Inverse kinematics solved all 300 G1+ and 200 WidowX test poses (median errors 1.1 and 0.35 mm).

Conclusions. A consumer smartphone can capture first-person video, head pose, and hand motion that can be mapped to actions humanoid and bimanual robots can execute. These findings demonstrate feasibility; they do not yet establish dataset quality or robot performance. Data collection, training, and deployment are the next steps.

Keywords: Egocentric video; Operating room; Foundation model; Hand tracking; Robotic surgery; Smartphone

Introduction

Robot learning is shifting from narrow, single-task policies toward generalist foundation policies whose ability to generalize depends on the breadth and diversity of their training data. Vision-language-action (VLA) models translate camera observations and natural-language instructions into robot actions [1–5]. More recently, world action models (WAMs) have built on video-generation models to predict future visual states and guide action generation [6, 7]. Both approaches rely primarily on vision to inform action, yet collecting sufficiently diverse training data remains a major bottleneck to real-world deployment. Teleoperation provides demonstrations on physical robots, but achieving broad coverage of tasks and environments requires substantial, coordinated collection efforts. The Open X-Embodiment collaboration illustrates the value of pooling these efforts: it assembled more than one million real-robot trajectories across 22 robot types and showed that models trained on pooled data could outperform those trained on data from a single robot [8].

Surgery has no comparable pool. Surgical data science has produced mainly endoscopic video [9, 10] and small kinematic datasets from dry-lab tasks on a da Vinci system [11]. Autonomy research has nonetheless made real progress on this thin base, including supervised autonomous anastomosis in pigs [12, 13], imitation-learned suturing and tissue tasks on the da Vinci Research Kit [14], and autonomous clipping and cutting during ex vivo cholecystectomy [15]. Our own voice-interactive robotic scrub nurse, ASIA, was built the same way, by fine-tuning a general-purpose robotic foundation model [4] on

teleoperated demonstrations [16]. Some of these systems learned from a single robot through teleoperation, and such demonstrations are slow and costly to collect.

The gap is wider than the data these systems use, because most work in the operating room (OR) never passes through a console: scrub technologists set up the back table, count sponges and instruments, and pass instruments; assistants retract, suction, irrigate and cut sutures; and in open surgery and microsurgery, surgeons operate with their own hands. No current dataset records this work from the point of view of the person doing it.

Outside medicine, egocentric (first-person) human video has become a practical substitute for robot demonstrations. Large head-worn camera datasets show that everyday manual activity can be captured at scale [17, 18], and hand-pose estimators recover 21 hand key points or a full hand mesh from a single image [19–21]. With these tools, several groups have trained robot policies directly from human hand motion, using ordinary video [22], a handheld gripper with a wrist camera [23], or egocentric headset recordings with hand tracking, including hundreds of hours with 3D hand tracking [24, 25]. Surgery has no equivalent source, which would require egocentric video, camera pose, hand pose and task labels captured under OR rules for sterility, consent and patient privacy.

Accordingly, we developed Surgical Intelligence Labs (SIL), a smartphone app that collects this data using the phone a clinician already owns. Our primary objective was to describe the design of the capture system, including its tasks, sensors, on-device processing and privacy controls. Our secondary objectives were to measure the feasibility of its sensing and of its retargeting to two robot embodiments, and to outline a roadmap to an OR foundation model that could control humanoid robots in the scrub technologist, assistant and surgeon roles and be adapted to console robots such as the da Vinci Xi and microsurgical robots such as Symani. This is a technical report and position paper that reports no patient data and no clinical outcomes.

Methods

Study design

We conducted a technical development and feasibility study. The app was built first, and its sensing and retargeting were then measured on test data. No patients or study participants were recorded. Head tracking was tested on one pilot recording, and hand tracking, three-dimensional (3D) hand fitting and retargeting on a 63-s recording of a suturing exercise, both made by the authors in a non-clinical setting. The sections below describe the app and what it records, its privacy safeguards, how hands and head motion are tracked, how that motion is mapped onto two robots (retargeting), and how each part was evaluated.

System overview

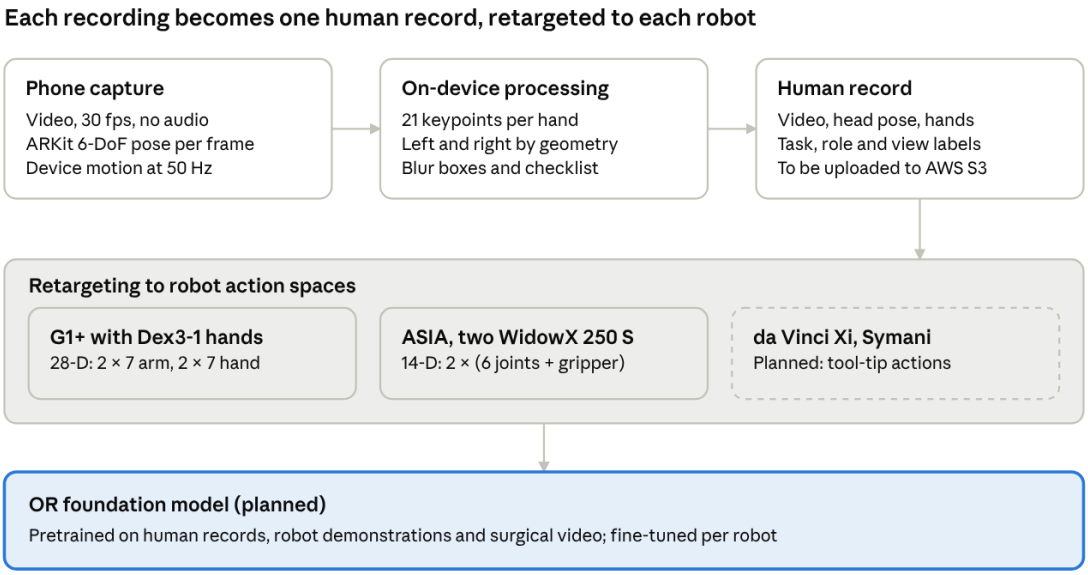


Fig. 1 Data pipeline. Capture, processing and retargeting to the two implemented robots run on the phone. Console and microsurgical retargeting and the foundation model are planned.

Fig. 1 shows the data pipeline: the phone records a clinician's view of a task, processes it into a record of hand and head motion, and converts that record into commands for robots. The app is built with Expo and React Native in TypeScript, and capture and hand tracking run in native iOS modules written in Swift. It currently runs on iPhone, and an Android version is planned. A four-screen onboarding sequence (Fig. 2) states the purpose of the app and the mount options, and it states that operating-room recording opens only after the contributor's institution has been approved.

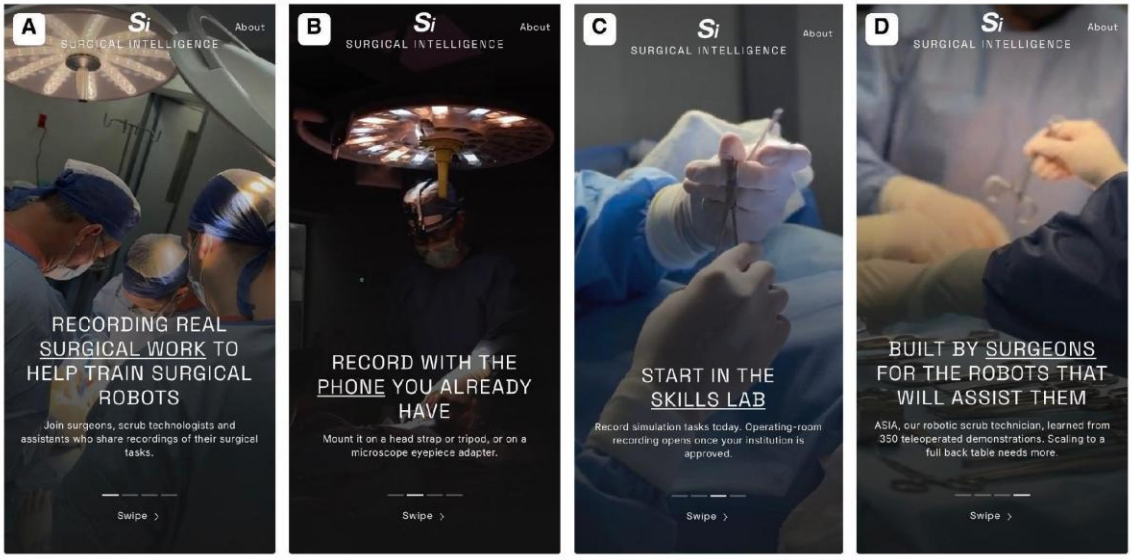


Fig. 2 Onboarding. (A) Purpose: recording real surgical work to help train surgical robots. (B) Recording with the contributor's own phone on a head strap, tripod or microscope eyepiece adapter. (C) Simulation tasks first;

operating-room recording opens after institutional approval. (D) The link to ASIA, the authors' robotic scrub technician [16], trained on 350 teleoperated demonstrations.

Tasks

Seven tasks cover the scrub technologist, assistant and surgeon roles and three imaging views: the laparoscopic monitor, the robotic console display and the operating microscope (Table 1, Fig. 3). Each task card gives step-by-step instructions, the allowed mounts and the allowed settings. Every task can be recorded in simulation, whereas the operating-room setting stays locked until the review team verifies the contributor's institution.

Table 1 Recording tasks.

Task	Role or view	Phone mount	What is recorded
Set Up the Back Table	Scrub technologist	Head	Instruments, drapes and supplies arranged on the back table and Mayo stand per the tray card
Pass Instruments	Scrub technologist	Head	Loading the needle driver, passing instruments handle-first, taking them back
Count Sponges & Instruments	Scrub technologist	Head	A standard count of sponges, sharps and instruments with the circulator
Open Surgery	Surgeon or assistant	Head	Suturing, retraction, suction and irrigation
Laparoscopic View	Laparoscopic monitor	Tripod or monitor	Box-trainer drills, such as peg transfer, or cases
Robotic Console View	Robotic vision-cart display	Tripod or monitor	Simulator exercises, dry-lab tasks or cases
Microscope View	Operating microscope	Eyepiece adapter or head	Vessel preparation and anastomosis, as surgeon or assistant

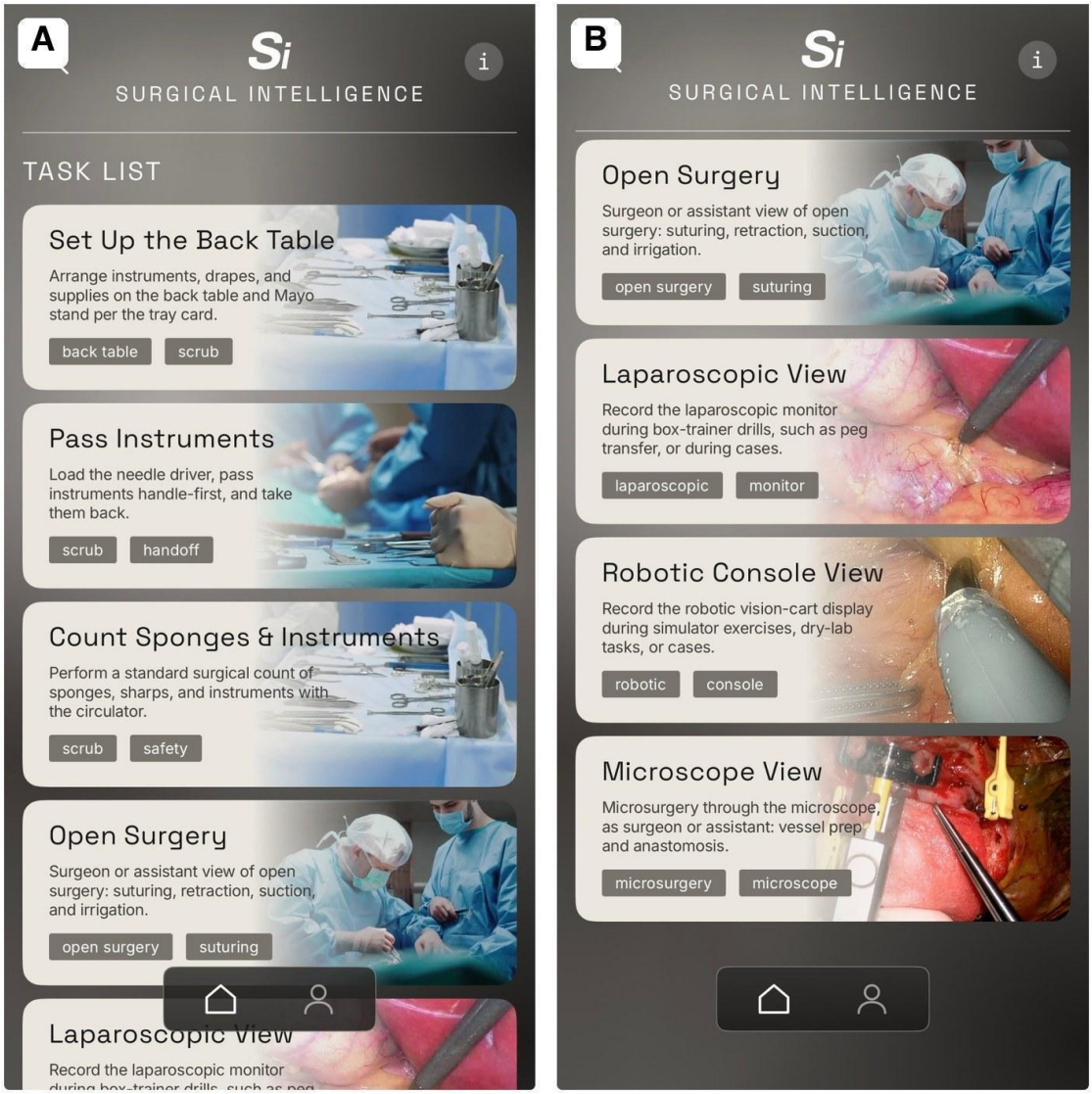


Fig. 3 Task list. (A) Back-table setup, instrument passing, counts and open surgery. (B) Open surgery and the three imaging views. Card images for the laparoscopic, robotic and microscope views are excerpts of CC BY 4.0 videos [26–28].

Capture

The app records through ARKit, Apple's augmented-reality framework for iPhone and iPad [29], whenever the phone supports world tracking (Fig. 4). Phones without ARKit support fall back to the standard camera and still log device motion. World tracking estimates the phone's position and orientation in 3D (its 6-degree-of-freedom, or 6-DoF, pose) for every camera frame. It does so by fusing features tracked in the camera image with the phone's motion sensors, a method known as visual-inertial odometry. In practice, this tells the app where the clinician's head is, and where it is pointing, throughout the video.

Each recording stores four kinds of data. The video is written frame by frame to an H.264 movie at 30 frames/s, at least 1,920 pixels wide and 14 Mbit/s, with no audio track. For the same frames, the app writes the phone's pose to a JSON-lines file (time, position in meters, orientation as a quaternion, tracking

state and the camera's lens intrinsics) on the same clock as the video, in a world frame aligned with gravity whose origin is where tracking started. Device motion (orientation, acceleration and rotation rate) is logged at 50 Hz, and metadata records the task, setting, mount, duration, date, phone model, operating-system version and the de-identification record. A recording stops at 60 minutes or 6 GB.

A head-mounted phone's screen cannot be read while it is worn, so the recording screen guides framing by text, touch and sound (Fig. 4). With a head mount, before recording starts, ARKit's camera orientation gives the tilt below the horizon four times a second and Apple Vision counts the hands in view twice a second. The screen asks the contributor to tilt the phone until it points 48–72° below the horizon and to hold their hands where they will work ("Hold your hands where you will work", "Only one hand in view"), and a haptic pulse confirms when both hands are in view. In the simulation setting, the phone also speaks: "Both hands in view", "Tilt the phone down" or "up" during the optional 5-or 10-s start delay, "Recording in 5 seconds", "Recording", "Minimum time reached" and "Recording stopped". Haptic pulses mark each second of the countdown, the start and the stop, and the stop button ends a recording at once.

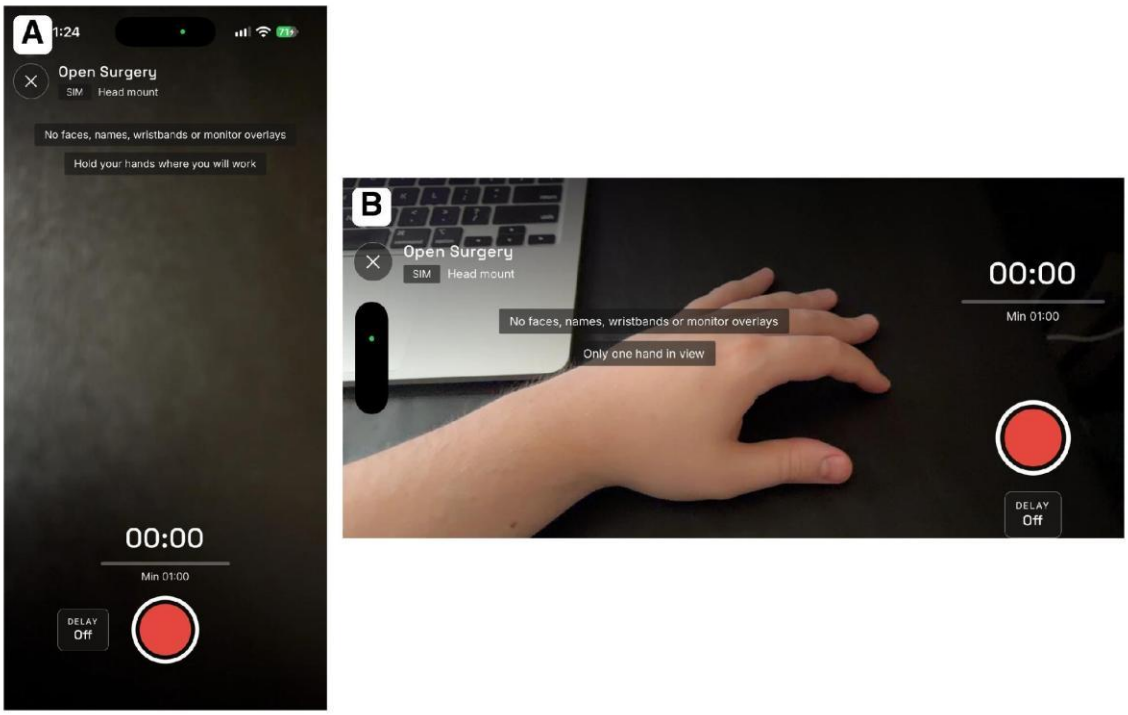


Fig. 4 Recording screen for the Open Surgery task in simulation (SIM) with a head mount, before recording. (A) With no hands in view, the framing guide asks the contributor to hold their hands where they will work. (B) In landscape, with one hand in view, it reports "Only one hand in view"; with both hands in view, the phone buzzes and, in simulation, says "Both hands in view". A banner reminds the contributor to keep faces, names, wristbands and monitor overlays out of frame. The screen also shows the timer with the task's minimum length (1 min), the start-delay button and the record button.

Privacy, consent and de-identification

Before the first recording, a one-time screen lists what is recorded and where it is kept and asks for an affirmative tap. The app tells contributors never to record patient faces, names, wristbands, charts or screens showing patient data, and to record in an operating room only after institutional approval and only where local policy allows. First-person tasks use a head mount only, and contributors follow their

facility's infection-control rules for the phone and mount. The app never records sound, so speech cannot carry identifiers.

On the review screen, the contributor draws blur boxes over any identifying region and completes a checklist modeled on the Health Insurance Portability and Accountability Act (HIPAA) Safe Harbor identifiers [30]. Every recording, from simulation and the operating room alike, will then be uploaded to an encrypted Amazon S3 quarantine bucket under a HIPAA Business Associate Addendum with Amazon Web Services (AWS). There, the contributor's blur boxes will be applied, and faces and on-screen text will be blurred automatically. Staff will review only the blurred version before a recording enters the dataset, and raw originals will be deleted after review. Simulation footage will follow the same path because it can still show faces, badges or screens.

Hand tracking

After recording, the contributor can run hand tracking on the phone. It uses Apple Vision's hand-pose request [31], which returns 21 keypoints per hand (points on the wrist, finger joints and fingertips). We sample at 15 frames/s and request up to four hands per frame. Off-the-shelf detectors are built for bare hands in clear view, so we added three steps to handle surgical footage.

First, the wearer's hands often enter from the bottom edge, cut off at the wrist, so each frame is analyzed twice: as recorded, and with a neutral grey margin of 25% on every side (edge padding). Detections from the padded pass are mapped back to the original frame and kept when their centers lie at least 0.1 (in normalized image units) from existing detections.

Second, Vision's left/right label is unreliable for gloved hands seen by the rear camera, so we assign sides ourselves. With two hands, the hand whose wrist is further left is the left hand. With one hand, a vote combines wrist position with thumb side (the sign of the cross product of the wrist-to-middle-knuckle and wrist-to-thumb vectors), and ties go to the nearest hand seen in the previous second.

Third, because the wearer's hands enter from the bottom of an egocentric frame, the two detections with the lowest wrists are kept, and gaps of up to 0.6 s are interpolated and flagged.

The output is a compact JSON file with 21 keypoints per hand per sampled frame.

Three-dimensional hand fitting

Vision returns image points only, so the app lifts each tracked hand to 3D by fitting an articulated hand model to it (skeleton fitting). The model has the bone lengths of a typical adult hand and 20 joint angles in the order of the Wuji Hand 2's joints (four per finger: knuckle flexion and spread, middle and end joints; four for the thumb). Its thumb turns about its own length as its base bends and sweeps, as the human thumb's saddle joint does, so that an opposed thumb curls toward the fingers. For each hand and frame, a Levenberg–Marquardt least-squares solver finds the hand's position, orientation and joint angles whose 21 points, projected through the lens intrinsics that ARKit logs with every frame, best match the tracked points. Soft joint limits, a pull toward the previous frame's solution and a coupling between each finger's middle and end joints keep the fit anatomically plausible and steady. Where the thumb tip overlaps a fingertip in the image, the fit prefers a pose in which they touch, because seen from above a touch and a near-miss project alike. Fits whose points land, on average, more than 4% of the image size from the tracked points are discarded, and those frames keep only their image points. The fit runs once per recording on the phone, after tracking, and is stored with the key points.

Head trajectory

From the pose file, the app derives head speed, path length, pitch (up-down tilt), yaw rate (how fast the head turns side to side) and height. Standing height is the 95th percentile of camera height, and crouch is the drop below it.

Retargeting to robot embodiments

Retargeting maps recorded human motion onto a robot's joints, keeping sides: the human left hand drives the robot's left hand.

For the Unitree G1+ humanoid with Dex3-1 hands, kinematics (the geometry of the arms and joints) come from the manufacturer's Unified Robot Description Format (URDF) files (unitree_ros, BSD 3-Clause) [32]. Each arm has seven joints (shoulder pitch, roll and yaw; elbow; and wrist roll, pitch and yaw), and each Dex3-1 hand has seven joints across three fingers. Each fitted hand sets a wrist target in the robot's pelvis frame: its 3D position and orientation relative to the eyes, turned into a level body frame with ARKit's head orientation, scaled to the G1+'s shorter arms (forward and vertical offsets by 0.7, sideways offsets unscaled) and kept within reach. The two wrists are kept at least 13 cm apart sideways, and where fingertips of the two hands come within 7 cm of each other, both wrists shift so that the robot's fingertips meet at the human's gap. Frames without a fit fall back to the hand's image position and apparent size. A damped least-squares inverse-kinematics solver [33], which finds the joint angles that place the wrist at that target, reaches it with joint-limit scaling and a null-space pull toward a rest pose, and finger flexion sets the Dex3-1 joints. Each frame therefore yields a 28-dimensional action (2 × 7 arm joints, 2 × 7 hand joints), and head pose sets torso lean, crouch and neck pitch.

The G1+ can instead carry five-finger Wuji Hand 2 hands, modeled from the manufacturer's URDF files (wuji-description, MIT License) [34]. Their 20 joints match the human hand's one to one: four per finger (knuckle flexion and spread, middle and end joints) and four for the thumb (two at its base, then its middle and end joints). Copying the fitted human angles onto these joints failed whenever the thumb touched a finger, because the robot's fingers are longer and its thumb axes are placed differently. We therefore retarget by position, as in vision-based teleoperation of dexterous hands [35]. A Levenberg–Marquardt solver finds the 20 joint angles that place each robot fingertip and end joint where the human's are, relative to the finger's knuckle and scaled by the robot-to-human finger length, and the thumb's knuckle, end joint and tip where the human's are, relative to the thumb's base. Where the human thumb tip comes within 6 cm of a fingertip, the robot's two tips are pulled to the same gap, unscaled below 2 cm, so that pinches and instrument grips carry over. A small pull toward the copied human angles settles what the positions leave free, the thumb's end joints are kept from bending backward, and each frame starts from the previous frame's solution. With these hands, each frame yields a 54-dimensional action (2 × 7 arm joints, 2 × 20 hand joints). Because the commercial hand is sold with a soft outer covering that its URDF omits, the app can also draw each link shrink-wrapped in a cover for display ("Skin").

ASIA consists of two WidowX 250 S arms, each with six joints and a parallel gripper [36]. A damped least-squares solver places the gripper tip, hand roll sets wrist rotation, and the thumb-to-index distance sets gripper opening. Each frame yields a 14-dimensional action (2 × 6 joints, 2 grippers) in the layout used by ALOHA [37]. All robots are rendered on the phone with three.js from simplified meshes of their URDFs (Fig. 5).

Evaluation

To test whether the robot models move correctly, we compared the G1+ forward kinematics (hand position computed from joint angles) at the zero configuration with the URDF's own forward kinematics.

We then tested inverse kinematics (finding joint angles for a desired hand position) on 150 random reachable poses per arm, drawing joint values from the middle 40% of each joint's range. The solver started from the true configuration perturbed by up to ±0.2 rad and ran 60 iterations, and success meant a position error under 5 mm. For the WidowX, we solved 200 random reachable gripper positions from the neutral pose with 80 iterations, and success meant an error under 1 mm. We report the success rate, the median and 95th-percentile error, and the mean time per solve (Node.js 22 on an Apple M1 laptop).

For head tracking, we report path length, mean speed and mean yaw rate from one pilot recording on an iPhone 17, and, on the suturing recording described below, the share of video frames with normal ARKit tracking and the phone's mean tilt below the horizon.

To test 3D fitting and retargeting, one author recorded 63 s of a suturing exercise on a practice pad with a head-mounted iPhone 17 (Open Surgery task, simulation setting), and hands were tracked and fitted on the phone. We report the share of sampled frames in which both hands were tracked and the share of hand observations whose 3D fit was kept. For every hand and frame in which the fitted thumb tip lay within 3 cm of a fingertip (pinches and instrument grips), we compared that gap on the fitted human hand with the same gap on the Wuji Hand 2, under position-based retargeting and under direct copying of the fitted joint angles. We also counted robot thumbs whose middle and end joints bent in opposite directions by more than 30° (kinked), and, in frames in which fingertips of the two hands came within 4 cm of each other, compared the cross-hand fingertip gap of the robot's placed arms with the human's. These measurements were run on the laptop with the app's own code.

Generative artificial intelligence disclosure

Generative artificial intelligence tools were used to assist with coding, debugging and validation of the application, and to assist with the grammar and clarity of the manuscript. All outputs were reviewed, approved and overseen by the authors, who take full responsibility for the content of the manuscript.

Results

A working capture-to-robot chain

The full chain runs on an iPhone 17, from capture through hand tracking, 3D fitting and retargeting to the robot view (Figs. 2–5). For each recording, the phone keeps the movie, the pose log, the device-motion log and the metadata, and, once tracking has run, the hand-keypoint file with its 3D fit. The bundled robot models hold 50 meshes and 45 actuated joints for the G1+, 10 meshes and 9 actuated joints for each WidowX 250 S, and 22 meshes and 20 actuated joints for each Wuji Hand 2, and together they take 3.9 MB.

Kinematics

We first checked that the robot models were geometrically faithful. At the zero configuration, the G1+ forward kinematics matched the URDF's own forward kinematics within 0.04 mm on every axis, and the WidowX home pose matched the URDF tool point exactly. Both inverse-kinematics solvers reached every test target (Table 2), and the G1+ targets included orientation as well as position. At 0.07–0.22 ms per solve on a laptop, both solvers fit comfortably within the 33 ms between frames at 30 frames/s, so they could keep pace with video, although times on the phone were not measured.

Table 2 Inverse-kinematics accuracy on random reachable targets.

Arm	Success threshold	Targets	Solved	Median error	95th percentile	Mean time per solve
G1+ left	5 mm	150	150 (100%)	1.04 mm	1.92 mm	0.11 ms
G1+ right	5 mm	150	150 (100%)	1.10 mm	1.87 mm	0.07 ms
WidowX 250 S	1 mm	200	200 (100%)	0.35 mm	0.92 mm	0.22 ms

Head tracking

In the pilot recording, ARKit tracked the phone and yielded a head path of 2.1 m, a mean speed of 0.36 m/s and a mean yaw rate of 17°/s, and every video frame had a matching pose row on the same clock. This single recording is descriptive and does not characterize head motion across tasks or users.

Hand tracking, 3D fitting and retargeting on a suturing recording

On the 63-s suturing recording, ARKit reported normal tracking for all 1,877 video frames, each with a pose row on the same clock, and the phone pointed a mean 55° below the horizon, inside the framing guide's 48–72° band; the head moved little while suturing (1.1 m path, 0.02 m/s mean speed). The phone tracked both hands in 910 of 939 frames sampled at 15 frames/s (97%), and the 3D fit was kept for 1,831 of 1,849 hand observations (99%). After fitting, position-based retargeting reproduced the thumb-to-fingertip gaps of pinches and grips on the Wuji Hand 2 within 2 mm of the fitted human gaps (medians), whereas copying the fitted joint angles left the robot's thumb 10–15 cm from the fingertip (Table 3). No robot thumb was kinked, and the robot's finger joints stayed within a median 8° of the fitted human angles. In the 26 frames in which fingertips of the two hands came within 4 cm of each other, the robot's placed hands reproduced the human gap within 1 cm in 20 (77%). Fitting and retargeting took about 12 ms per hand on the laptop; times on the phone were not measured.

Table 3 Thumb-to-fingertip gaps in pinches and grips on the suturing recording (medians over hand-frames in which the fitted human gap was under 3 cm).

Thumb to	Hand-frames	Human (fitted)	Robot, angles copied	Robot, position-based	Robot, worst frame
Index finger	1,620	1.9 cm	10.5 cm	1.8 cm	3.5 cm
Middle finger	1,226	1.8 cm	12.3 cm	1.9 cm	4.9 cm
Ring finger	382	2.6 cm	14.8 cm	2.8 cm	3.5 cm
Little finger	21	2.7 cm	11.9 cm	2.9 cm	3.4 cm

Retargeting in the Collect view

The Collect view (Fig. 5) shows a recording with its tracked hands, head metrics and the robot it drives. On the suturing recording, both robot arms follow the author's hands, and with the Wuji Hand 2 each robot finger follows the corresponding human finger (Fig. 5A, B). A toggle switches the same record between the G1+ with Dex3-1 hands, the G1+ with Wuji Hand 2 hands, bare or under their cover, and ASIA, whose two arms follow the hands' positions and whose grippers follow the thumb-to-index openings (Fig. 5C). Left (L) and right (R) labels beside each robot hand confirm that sides are kept. The

sample that the app shows new contributors is a 5-s excerpt of the same recording, with its tracked hands and ARKit head pose.

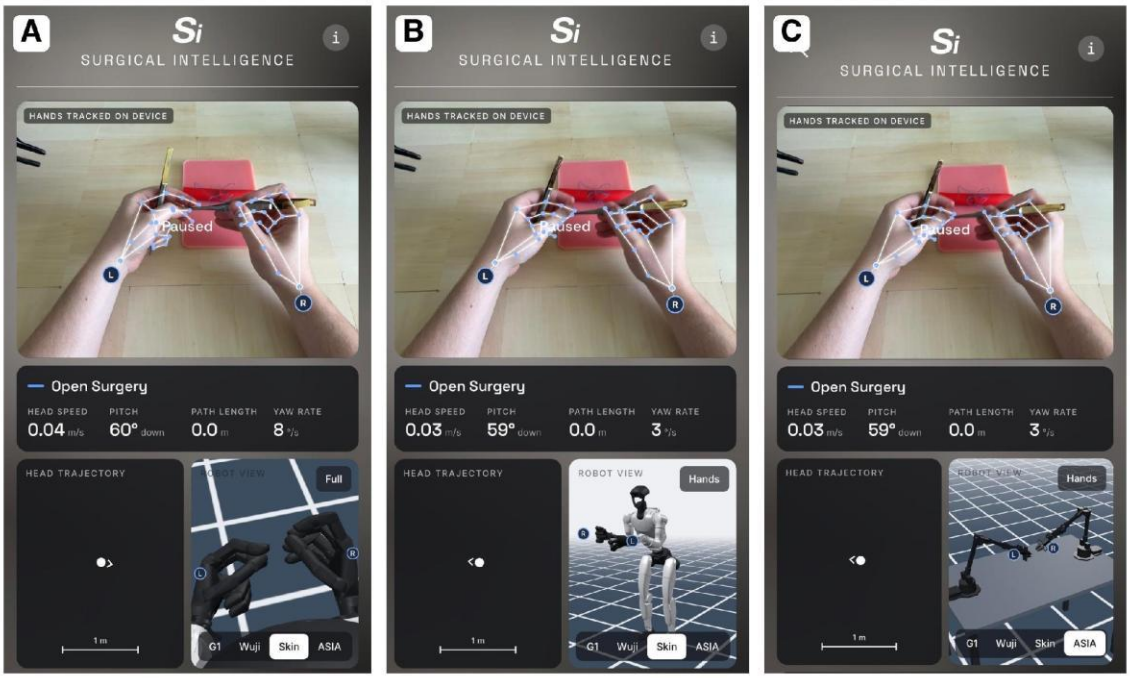


Fig. 5 Collect view on the suturing recording (head-mounted iPhone 17, simulation setting). Top: video with both hands tracked on the phone (21 keypoints each, L and R). Middle: head metrics from ARKit. Bottom right: the robot driven by the tracked hands, chosen with the G1 / Wuji / Skin / ASIA toggle. (A) Close-up of the G1+'s Wuji Hand 2 hands, drawn under their soft cover, each finger following the same finger of the author. (B) The full G1+ humanoid with the same hands. (C) ASIA, two WidowX 250 S arms, whose grippers follow the hands' positions and thumb-to-index openings.

Discussion

Principal findings

This study evaluated a smartphone system for capturing first-person operating-room data and mapping human motion to robot models. Using a phone clinicians already own, the system captured three complementary data streams for robot learning: first-person video, head pose synchronized with the video, and hand motion, under privacy controls designed for the operating room. Sensing was feasible in the settings tested, and robot models could reach estimated hand-position targets with approximately 1 mm of positional error. On a suturing exercise recorded by one author, both hands were tracked in nearly every frame, and once lifted to 3D, the hand motion carried onto a five-finger robot hand finger by finger, with pinches and instrument grips reproduced within millimeters of the fitted human hand.

Hand tracking is the component that matters most, because the hands are the data. Both hands were tracked in nearly every frame of the suturing recording, but it was made with bare hands on a clean field. In the operating room, gloves, blood, instruments covering the fingers and hands cut off at the frame edge will make tracking harder; the app's edge padding and side assignment were built for those conditions but have not yet been tested there. These results show feasibility, not a dataset, and the value of the app will depend on how many recordings it gathers, from how many clinicians, across how many tasks.

Why egocentric data from the operating room

Egocentric recording has three practical advantages. It reaches roles that console logs never see: the scrub technologist, the assistant, the open surgeon and the microsurgeon. It needs no robot, because each recording captures an expert doing the real task in the real setting. And because contributors will be paid per minute of video that passes review, cost scales with usable data rather than with robot hardware or teleoperation time. Outside surgery, co-training on egocentric human data has improved robot policies [23–25].

Head pose, meaning where the head is and which way it points, matters as much as the video. Because the camera moves with the head, a model without the camera's pose cannot tell head motion from hand motion, and hand trajectories cannot be placed in a fixed world frame. ARKit supplies that pose for every frame with no extra hardware. Head direction is also a rough proxy for where the wearer is attending, which for an assistant robot is a cue to what the surgeon needs next.

From data to an operating-room foundation model

Our own robot illustrates both the promise and the limit of the current approach. ASIA is a voice-interactive robotic scrub nurse that we developed using teleoperation training: a bimanual platform of two robotic arms was fine-tuned from a pretrained vision-language-action model ($\pi_{0.5}$) on 350 demonstrations that a human operator performed by remote control [4, 16]. In a simulated tray environment, ASIA completed 81% of 100 autonomous instrument handoff and take-back trials and interpreted spoken commands reliably. Its success fell from 92% when instruments were laid out as in training to 70% when they were laid out in new positions, and its main failure was a missed grasp [16]. A few hundred demonstrations thus taught one robot a few tasks in one setting, and performance dropped when the setting changed. Broader and cheaper data, of the kind recorded here, is one route to closing that gap.

We see this app as one of four data sources for an OR foundation model, a single model trained on many kinds of data so that it can be adapted to many robots and tasks: egocentric human records from this app (video, head pose, hands and task labels); existing endoscopic and robotic-console video, which shows tool motion but carries no robot actions; teleoperated robot demonstrations, such as those behind ASIA [16] and those recorded on the da Vinci Research Kit [14, 15]; and simulation.

A natural architecture follows recent vision-language-action models [3, 5], which read images and a written or spoken instruction and output robot movements, and ASIA already uses one of them [4, 16]. A pretrained vision-language backbone would read the scene and the instruction, and an action module would generate short chunks of future actions by flow matching or diffusion, two generative methods for producing smooth sequences of movement [3, 38]. Human hands would be treated as one more embodiment with its own action head, as in EgoMimic [24], and pretraining would span all embodiments, as in Open X-Embodiment and Octo [8, 39]. Each robot would then need only a smaller set of its own demonstrations for fine-tuning. Instructions would come from the task cards and from annotated step labels; because the app records no audio, spoken requests such as "hold" or "cut" would have to be annotated rather than heard.

The three target roles differ in difficulty and risk. The scrub technologist role is the most realistic first target: back-table setup, counts and instrument handoffs are structured and repetitive and involve no tissue contact, and ASIA was built for them [16]. The assistant role, which includes retraction, suction, irrigation, suture cutting and camera holding, demands anticipation of the surgeon's needs, and the surgeon's head pose and hand motion carry the timing cues an assistant robot would need. The surgeon role carries the highest risk, and autonomy would be limited to well-defined sub-steps under supervision.

Deployment targets

Deployment would differ by platform. A G1-class humanoid, a robot with a torso, arms and hands shaped like a person's, shares the human body plan, so hand records map onto it directly, as shown here. Its limits are arm precision, payload, sterile draping and safety around staff, and its first jobs are likely to be away from the patient: the back table, instrument handling and counts.

For console robots such as the da Vinci Xi, in which the surgeon operates remotely from a console, the action space (the set of movements the robot can be commanded to make) is instrument-tip pose, jaw angle and camera motion, not hands. At the console, the surgeon's hand motion drives the instrument tips through scaled master-slave control, so hand trajectories transfer in principle. The app's console-view recordings add visual context but no robot kinematics. Earlier da Vinci generations are open to research through the da Vinci Research Kit [40], where imitation-learned surgical policies have already run [14, 15]. The Xi has no comparable open interface, so deployment there would require a partnership with the manufacturer and its own regulatory path.

Symani is a microsurgical robot that adds motion scaling and tremor filtering to two wristed micro-instruments and is in clinical use for lymphatic surgery [41]. Microscope-view recordings and hand-held microsurgical motion, such as needle handling and knot tying, could map to its instruments after scaling. Autonomy for sub-steps such as needle passing, under the surgeon's supervision, is a plausible first goal.

Safety, regulation and ethics

Autonomy should rise in stages, from robot assistance, in which the surgeon remains in control, to supervised autonomy, in which the robot performs defined steps under a surgeon's oversight, following the levels described by Yang et al. and Haidegger [42, 43]. An OR foundation model that controls a robot would be regulated as a medical device, and each deployment would need validation for its intended use, platform and level of autonomy. Data governance must keep pace: contributor consent, institutional approval, de-identification before any upload, and the contributor's right to delete.

Limitations

Several limitations should temper interpretation. First, this was a feasibility study: we tested one phone model and two recordings made by the authors in a non-clinical setting, collected no dataset and trained no model, and the retargeting measurements compare the robot with the fitted human hand rather than with measured ground truth. Second, hand depth is inferred, not measured. Vision returns image keypoints, and the 3D fit assumes fixed adult hand proportions, so its depth and scale are approximate; metric 3D hand pose needs either a learned model such as HaMeR [20], whose MANO hand model [21] carries license limits on commercial use, or LiDAR depth on Pro-model iPhones, which we have not yet used. Third, our recordings had bare hands on a clean simulation field; in the operating room, gloves, blood, instruments covering the fingers and hands cut off at the frame edge will make hand tracking harder, and the app's edge padding and side assignment have not yet been tested there. Fourth, an embodiment gap remains. Forces, tissue feel and tool-tissue contact are not observed, robot hands differ from human hands, and our retargeting matches fingertip positions, not forces. Fifth, a head-mounted phone may change posture and is not appropriate in every OR, and contributors self-select. Finally, staff appear in view, so consent and institutional policy must cover everyone in the room, not only the patient.

Future directions

Future work should proceed in three steps. First, data collection should be approved by an institutional review board, beginning in simulation labs and then moving to operating-room pilots at approved institutions. Second, the fitted 3D hands should become metric, from LiDAR depth on Pro-model phones

or learned 3D hand models, and be paired with per-step labels. Third, recordings should be exported to an open robot-learning format such as LeRobot [44] to train baseline policies, with ASIA on back-table tasks as the first target.

Conclusion

We developed Surgical Intelligence Labs, a smartphone app that turns a clinician's own iPhone into an egocentric data-collection device for the operating room. The app records silent video, per-frame head pose and device motion, tracks hands on the phone and fits them in 3D, and keeps everything local under consent and de-identification controls. Its records already drive two robots, a humanoid with dexterous five-finger hands and a bimanual scrub-technician cell, with millimeter-level kinematic accuracy, and pinches and grips carry over to the robot's fingers. Both hands were tracked in nearly every frame of a suturing exercise; gloved, blood-stained operating-room footage remains to be tested. These findings establish the feasibility of capture and retargeting; they do not establish dataset quality, policy performance or clinical benefit. The next phases are approved data collection, metric 3D hands and baseline policies trained on the resulting dataset, as steps toward a foundation model that could serve the scrub technologist, the assistant and the surgeon on humanoid, console and microsurgical robots.

References

1. Brohan A, Brown N, Carbajal J, et al. RT-2: Vision-language-action models transfer web knowledge to robotic control. In: Proceedings of the 7th Conference on Robot Learning (CoRL). PMLR 229; 2023. arXiv:2307.15818.
2. Kim MJ, Pertsch K, Karamcheti S, et al. OpenVLA: An open-source vision-language-action model. In: Conference on Robot Learning (CoRL); 2024. arXiv:2406.09246.
3. Black K, Brown N, Driess D, et al. π_0: A vision-language-action flow model for general robot control. arXiv:2410.24164; 2024.
4. Physical Intelligence, Black K, Brown N, et al. $\pi_{0.5}$: A vision-language-action model with open- world generalization. arXiv:2504.16054; 2025.
5. NVIDIA, Bjorck J, Castañeda F, et al. GR00T N1: An open foundation model for generalist humanoid robots. arXiv:2503.14734; 2025.
6. Ye S, Ge Y, Zheng K, et al. World action models are zero-shot policies. arXiv:2602.15922; 2026.
7. Black Forest Labs. FLUX 3 Action: A 7B world action model for robot control. September 2026. https://bfl.ai/blog/flux-3-action. Accessed October 26, 2026.
8. Open X-Embodiment Collaboration, O'Neill A, Rehman A, et al. Open X-Embodiment: Robotic learning datasets and RT-X models. In: IEEE International Conference on Robotics and Automation (ICRA); 2024. arXiv:2310.08864.
9. Maier-Hein L, Vedula SS, Speidel S, et al. Surgical data science for next-generation interventions. Nat Biomed Eng. 2017;1(9):691–696.
10. Twinanda AP, Shehata S, Mutter D, Marescaux J, de Mathelin M, Padoy N. EndoNet: A deep architecture for recognition tasks on laparoscopic videos. IEEE Trans Med Imaging. 2017;36(1):86–97.

11. Gao Y, Vedula SS, Reiley CE, et al. JHU-ISI Gesture and Skill Assessment Working Set (JIGSAWS): A surgical activity dataset for human motion modeling. In: MICCAI Workshop on Modeling and Monitoring of Computer Assisted Interventions (M2CAI); 2014.
12. Shademan A, Decker RS, Opfermann JD, Leonard S, Krieger A, Kim PCW. Supervised autonomous robotic soft tissue surgery. Sci Transl Med. 2016;8(337):337ra64.
13. Saeidi H, Opfermann JD, Kam M, et al. Autonomous robotic laparoscopic surgery for intestinal anastomosis. Sci Robot. 2022;7(62):eabj2908.
14. Kim JW, Zhao TZ, Schmidgall S, et al. Surgical Robot Transformer (SRT): Imitation learning for surgical tasks. In: Conference on Robot Learning (CoRL); 2024. arXiv:2407.12998.
15. Kim JW, et al. SRT-H: A hierarchical framework for autonomous surgery via language- conditioned imitation learning. Sci Robot. 2025;10:eadt5254.
16. Khan M, Bombardelli J. Embodied AI in the operating room: a voice-interactive robotic scrub nurse for surgical instrument handoffs. J Robot Surg. 2026;20(1):894. doi:10.1007/s11701-026-03866-9. PMID: 42658368.
17. Grauman K, Westbury A, Byrne E, et al. Ego4D: Around the world in 3,000 hours of egocentric video. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2022.
18. Damen D, Doughty H, Farinella GM, et al. Scaling egocentric vision: The EPIC-KITCHENS dataset. In: European Conference on Computer Vision (ECCV); 2018.
19. Zhang F, Bazarevsky V, Vakunov A, et al. MediaPipe Hands: On-device real-time hand tracking. arXiv:2006.10214; 2020.
20. Pavlakos G, Shan D, Radosavovic I, Kanazawa A, Fouhey D, Malik J. Reconstructing hands in 3D with transformers. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR); 2024.
21. Romero J, Tzionas D, Black MJ. Embodied hands: Modeling and capturing hands and bodies together. ACM Trans Graph. 2017;36(6):245.
22. Qin Y, Wu YH, Liu S, et al. DexMV: Imitation learning for dexterous manipulation from human videos. In: European Conference on Computer Vision (ECCV); 2022.
23. Chi C, Xu Z, Pan C, et al. Universal Manipulation Interface: In-the-wild robot teaching without in-the-wild robots. In: Robotics: Science and Systems (RSS); 2024.
24. Kareer S, Patel D, Punamiya R, et al. EgoMimic: Scaling imitation learning via egocentric video. arXiv:2410.24221; 2024.
25. Hoque R, Huang P, Yoon DJ, Sivapurapu M, Zhang J. EgoDex: Learning dexterous manipulation from large-scale egocentric video. arXiv:2505.11709; 2025.
26. Zhou Y, Xiao L, Luo Z, Luo H, Tan Z, Wang T. Cystic plate approach in laparoscopic cholecystectomy: a consecutive retrospective analysis. Front Surg. 2024;11:1487568. doi:10.3389/fsurg.2024.1487568.
27. Butnari V, Mansuri A, Hanson M, et al. Robotic sigmoid colectomy with intracorporeal anastomosis: IMV first approach, a video vignette. Tech Coloproctol. 2026;30:51. doi:10.1007/s10151-025-03249-6.

28. Ganry L. Techniques useful in complex microvascular anastomosis. Int Microsurg J. 2021;5(2):2. doi:10.24983/scitemed.imj.2021.00151.

29. Apple Inc. ARWorldTrackingConfiguration. ARKit, Apple Developer Documentation. https://developer.apple.com/documentation/arkit/arworldtrackingconfiguration. Accessed October 26, 2026.

30. Office for Civil Rights, U.S. Department of Health and Human Services. Guidance regarding methods for de-identification of protected health information in accordance with the Health Insurance Portability and Accountability Act (HIPAA) Privacy Rule. 2012.

31. Apple Inc. VNDetectHumanHandPoseRequest. Vision, Apple Developer Documentation. https://developer.apple.com/documentation/vision/vndetecthumanhandposerequest. Accessed October 26, 2026.

32. Unitree Robotics. unitree_ros: robot description files for the G1 and Dex3-1 (BSD 3-Clause License). https://github.com/unitreerobotics/unitree_ros.

33. Wampler CW. Manipulator inverse kinematic solutions based on vector formulations and damped least-squares methods. IEEE Trans Syst Man Cybern. 1986;16(1):93–101.

34. Wuji Technology. wuji-description: robot description files for the Wuji Hand 2 (MIT License). https://github.com/wuji-technology/wuji-description. Accessed October 28, 2026.

35. Handa A, Van Wyk K, Yang W, et al. DexPilot: Vision-based teleoperation of dexterous robotic hand-arm system. In: IEEE International Conference on Robotics and Automation (ICRA); 2020:9164–9170.

36. Trossen Robotics. interbotix_ros_manipulators: description files for the WidowX 250 S (BSD 3-Clause License). https://github.com/Interbotix/interbotix_ros_manipulators.

37. Zhao TZ, Kumar V, Levine S, Finn C. Learning fine-grained bimanual manipulation with low- cost hardware. In: Robotics: Science and Systems (RSS); 2023.

38. Chi C, Feng S, Du Y, et al. Diffusion Policy: Visuomotor policy learning via action diffusion. In: Robotics: Science and Systems (RSS); 2023.

39. Octo Model Team, Ghosh D, Walke H, et al. Octo: An open-source generalist robot policy. In: Robotics: Science and Systems (RSS); 2024.

40. Kazanzides P, Chen Z, Deguet A, Fischer GS, Taylor RH, DiMaio SP. An open-source research kit for the da Vinci Surgical System. In: IEEE International Conference on Robotics and Automation (ICRA); 2014:6434–6439.

41. Lindenblatt N, Grünherz L, Wang A, et al. Early experience using a new robotic microsurgical system for lymphatic surgery. Plast Reconstr Surg Glob Open. 2022;10(1):e4013.

42. Yang GZ, Cambias J, Cleary K, et al. Medical robotics—Regulatory, ethical, and legal considerations for increasing levels of autonomy. Sci Robot. 2017;2(4):eaam8638.

43. Haidegger T. Autonomy for surgical robots: Concepts and paradigms. IEEE Trans Med Robot Bionics. 2019;1(2):65–76.

44. Cadene R, Alibert S, Soare A, et al. LeRobot: State-of-the-art machine learning for real-world robotics in PyTorch. 2024. https://github.com/huggingface/lerobot.