A Worn Skill-Capture System: End-to-End Hardware, SLAM, and Policy

Cornelius Gruss
Department of Mechanical Engineering
Boston University
Boston, MA, USA
cgruss@bu.edu

***Abstract*—We present a four-month solo reimplementation of Sunday Robotics' published Skill Capture Glove form factor that records human manipulation demonstrations as training data for imitation learning: 6-DoF pose, raw Hall joint proprioception, and time-of-flight ranging, synchronized to monocular GoPro Hero 10 video. The primary sensing contribution is a force-correlated Hall B_z signal recovered from the deliberate axial compliance of each PIP joint; a controlled scale-press experiment yields $+1.29$ counts/g ($R^2 = 0.64$) on the 3F PIP joint, with a flat residual on the rigid Index MCP joint as a negative control. After evaluating eight pose-tracking backends against measured failure modes (Sec. V), the production stack is Hierarchical-Localization with a Gaussian smoother at $\sigma = 5$ frames; a controlled AprilGrid validation places HLoc's 8.3 mm $p95$ error within 0.2 mm of the AprilGrid PnP $p95$ noise floor of 8.1 mm on static no-cup frames, locating remaining error on these regimes in capture-side hardware rather than algorithmic work. An ACT policy trained on 182 mug-on-coaster demonstrations reaches 28.8 mm position L_1, 6.31° rotation L_1, and 736 LSB Hall L_1 on a held-out validation set against the production HLoc + σ=5 ground truth—roughly $3.5\times$ the truth-source's own measurement floor. Code and a reproducible CLI for the full pipeline are available at https://github.com/corneliusgruss/skill-capture-glove.**

I. Introduction

Imitation learning for dexterous manipulation is bottlenecked by demonstration data. Robot-supervised teleoperation produces high-quality demonstrations but at low throughput; the natural alternative is to put the capture hardware directly on the human and record manipulation as it happens. The Universal Manipulation Interface (UMI) [1] demonstrated that a handheld gripper with a wrist-mounted fisheye GoPro running monocular-inertial SLAM can produce training-ready data at the throughput needed to scale beyond ALOHA-class teleoperation [2]. What public open-source handheld-capture stacks lack is *per-joint proprioception*—particularly squeeze force, which is what distinguishes "the hand reached the cup" from "the hand grasped the cup." Sunday Robotics, co-founded by UMI first author Cheng Chi and ALOHA/ACT first author Tony Zhao, has publicly demonstrated a worn Skill Capture Glove [3] that captures sensorimotor demonstrations from human hands, but the sensor architecture is not publicly documented and the system is not open-sourced.

This work presents a solo four-month reimplementation of Sunday's published form factor that adds the missing measurement layer. The specific contributions are:

1) **End-to-end integrated capture stack.** A worn glove (V3) with custom DAQ PCB; a continuous master-tape recorder; a per-demo offline ingest pipeline that aligns GoPro Hero 10 video with proprioceptive streams to within a video frame; a SLAM stage that produces per-frame 6-DoF pose in a gravity-aligned tag world frame; and an alignment stage that emits a single training-ready Parquet file per demonstration.
2) **Hall-effect joint proprioception with a force-correlated B_z signal on compliant joints.** Each MLX90393 sensor delivers three magnetic-flux channels: B_x and B_y give continuous joint angle via $\mathrm{atan2}$; the B_z channel changes monotonically with squeeze force *because* the PIP joints carry a deliberate axial slit that compresses under fingertip load. A controlled scale-press experiment recovers $+1.29$ counts/g ($R^2 = 0.64$, 300–1300 g) on the compliant 3F PIP joint, with a flat residual on the rigid Index MCP joint as a negative control.
3) **Pose tracking proven against ground truth.** Eight pose-tracking backends were tried and discarded over four months (Sec. V); the production stack is Hierarchical-Localization (SuperPoint + LightGlue + NetVLAD + COLMAP) [4]–[8] with a pose-graph batch smoother and a final Gaussian filter at $\sigma = 5$ frames. A controlled AprilGrid validation experiment shows HLoc's 8.3 mm $p95$ error to be within 0.2 mm of the AprilGrid PnP $p95$ noise floor on static no-cup frames—bounding further algorithmic work as unlikely to improve the metric on these regimes and pointing the next improvement at capture-side hardware. Fast-motion behavior is not separately bounded (Sec. VIII).
4) **Offline policy training on 182 demonstrations.** An ACT policy [2] reaches **28.8 mm position L_1, 6.31° rotation L_1, and 736 LSB Hall L_1** on a held-out validation set against the production HLoc + σ=5 ground truth. Final metrics are reported after trajectory-quality filtering (Sec. VII-D).

This report covers the capture pipeline and offline policy training. Actuated deployment, bimanual capture, and the V4 hardware revisions discussed in Sec. VIII are out of scope.

The remainder of the report is organized as follows. Sec. II places this work against UMI, ALOHA, Diffusion Policy, and

HLoc. Sec. III gives a one-figure system overview. Sec. IV documents the glove hardware including the Hall force-sensing experiment. Sec. V covers pose tracking and AprilGrid validation. Sec. VI covers time and spatial synchronization. Sec. VII covers policy training and dataset quality control. Sec. VIII lists honest limitations; Sec. IX concludes.

II. Related Work

Diffusion Policy and ACT. Chi et al. [9] introduce Diffusion Policy, which models the action distribution as a denoising diffusion process per timestep conditioned on visual observation. Zhao et al. [2] introduce the Action-Chunked Transformer (ACT) for the ALOHA bimanual platform: a CVAE encoder over a multi-step action chunk and a DETR-style decoder, with a two-frame observation horizon. We adopt ACT in Sec. VII in preference to Diffusion Policy for two reasons. First, the mug-on-coaster task routinely has the coaster out of frame mid-pickup; a long action chunk (300 frames, 5 s at 60 fps in our setting) lets the policy plan past such gaps despite the two-frame observation horizon. Second, with 182 demonstrations and a single task, ACT's deterministic chunked prediction is more sample-efficient than per-step diffusion sampling; a $\sim 10\times$ larger demonstration set with multimodal task variants would make Diffusion Policy the natural swap.

UMI and successors. Chi et al. [1] introduce the Universal Manipulation Interface, a handheld gripper with a wrist-mounted fisheye GoPro running monocular-inertial SLAM that captures human demonstrations at training-relevant throughput. UMI is the closest published precedent to this work and is the source of the Hero 10 + ORB-SLAM3 + Docker stack we adopted in Sec. V. Subsequent UMI-family work has uniformly replaced ORB-SLAM3 [10] with newer SLAM stages, which informed our pivot from ORB-SLAM3 to Hierarchical-Localization in Sec. V.

ALOHA. Zhao et al. [2] introduce ALOHA, a low-cost teleoperated bimanual platform with 6-DoF arms; ACT was developed against ALOHA and our Sec. VII architecture inherits directly from that codebase.

Sunday Robotics' Skill Capture Glove. Sunday Robotics, co-founded by UMI first author Cheng Chi and ALOHA/ACT first author Tony Zhao, has publicly demonstrated [3] a worn skill-capture glove in 2025. The internal architecture is not published; this work reimplements the form factor from public photographs and product descriptions, and the choice of MLX90393 Hall sensors plus the force-correlated B_z experiment in Sec. IV-C are designed contributions of this work, not claims about Sunday's stack.

Hierarchical-Localization (HLoc). Sarlin et al. [4] introduce a coarse-to-fine visual-localization paradigm and the HF-Net retrieval-then-matching architecture that the modern HLoc framework inherits. The implementation we use pairs NetVLAD [7] image retrieval with SuperPoint [5] keypoints and LightGlue [6] matching to produce 2D–3D correspondences, then solves per-frame PnP against a pre-built COLMAP [8] SfM model. We adopt this stack in Sec. V after diagnosing that ORB-SLAM3's reproduction reliability is bounded by closed-source map-loading code we cannot tune.

III. System Overview

The capture-to-training pipeline runs in five discrete stages with stable on-disk handoffs (Fig. 1).

(1) Capture. Two recorders run independently. The GoPro Hero 10 records 60 fps video to its own SD card; a host-side master-tape recorder writes per-day Parquet files for proprioception (Hall + ToF at 30 Hz) and IMU ($\sim$680 Hz observed) from the DAQ board over USB. GoPro Labs UTC firmware (Sec. VI) ensures both clocks share a common absolute time base.

(2) Offload and ingest. After a session, MP4s are copied from the SD card to local disk; ingestion then creates a per-demo session directory, slicing the master-tape Parquet to the recording window and writing per-session metadata.

(3) SLAM. Mapping clips first run an ArUco-tag detector per frame, then build a SfM model via SuperPoint + LightGlue + NetVLAD + COLMAP through Hierarchical-Localization (Sec. V); a tag-frame calibration step recovers the transform that gravity-aligns the world. Demo clips localize against the workspace map atlas to produce a per-frame camera trajectory.

(4) Align. The alignment stage interpolates the proprioceptive streams onto the GoPro frame timeline and emits `episode.parquet`, a single Parquet file with the schema consumed directly by the policy DataLoader: 6-DoF MTCP pose, 6-DoF camera pose, 12 raw Hall channels, two ToF channels, and per-frame tracking flags.

(5) Train. Policy training runs on the BU Shared Computing Cluster, reads `episode.parquet` files via a manifest, and writes wandb-tracked checkpoints back to project storage.

By design, the SLAM-backend swaps documented in Sec. V do not propagate to the policy DataLoader: the alignment stage absorbs each backend change without altering the downstream `episode.parquet` schema. A scrubbable dashboard that overlays video, trajectory, and proprioception on a shared time axis surfaced both the silent SLAM re-anchoring artifacts of Sec. VII-D and the encoder-pipeline delay of Sec. VI by direct visual inspection.

IV. Hardware

A. Glove form factor

The glove (V3) reimplements Sunday Robotics' published Skill Capture Glove form factor. Two finger assemblies—a single index finger and a three-finger unit—articulate at PIP and MCP joints. Four joints across the two assemblies (Index PIP, Index MCP, 3F PIP, 3F MCP) are Hall-instrumented: each carries a 3×2 mm diametric magnet read by a single MLX90393 sensor on a thin (1.2 mm) FR4 sensor PCB. The PIP joints are the load-bearing element of the mechanical design: each PIP carries a deliberate axial slit that gives the joint a known, tuned compliance under fingertip pressure. The fingertip caps transfer external load through the slit geometry rather than around it, which is the design intent that Sec. IV-C validates with measured force.

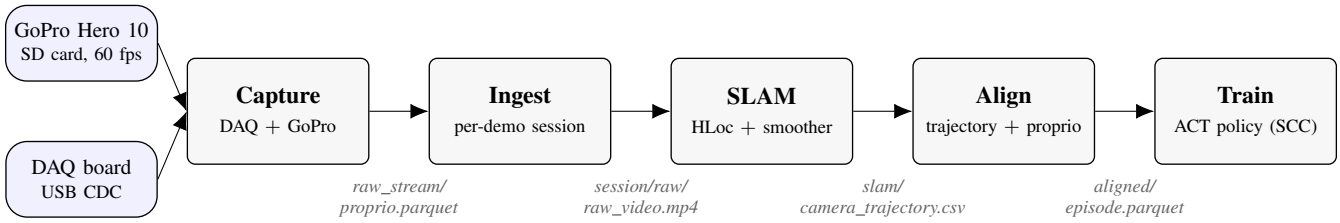
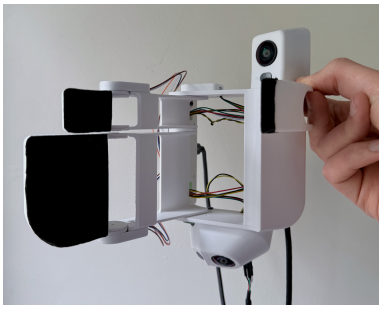


Fig. 1. System pipeline. Five stages with stable on-disk handoffs between them. Inputs (left): the GoPro Hero 10 records to its own SD card; the DAQ board streams proprioception and IMU over USB. Both are timestamped against a common UTC reference (Sec. VI). The SLAM stage changed multiple times during the project (Sec. V); the inter-stage schema stayed constant.



(a) Glove V3, this work

(b) Sunday Robotics, sunday.ai

Fig. 2. Form-factor comparison. Both share two finger assemblies (single index + three-finger unit), rigid knuckle housings, and top/bottom camera placement. The OV9782 cameras on V3 were the original on-glove SLAM sensors; the production stack uses an externally-mounted GoPro Hero 10 (Table I, row 6).

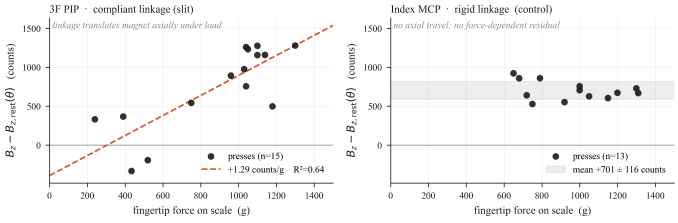


Fig. 3. Hall B_z residual versus applied scale-press force. Compliant 3F PIP joint gives a linear fit of $+1.29$ counts/g, $R^2 = 0.64$ across 300–1300 g. Rigid Index MCP joint shows a flat residual under the same load profile, confirming the design intent that PIP slits, not joint contact, transduce force into magnet translation.

B. DAQ PCB

The custom DAQ PCB carries an STM32G431 microcontroller, a TCA9548A I^{2}C multiplexer, a QMI8658C IMU, four MLX90393 Hall sensors (one per joint sensor PCB on the fingers), and two VL53L1X time-of-flight sensors. It exposes a single USB CDC link to the host, streaming proprio at 30 Hz (interpolated to the 60 fps GoPro frame timeline at align time) and IMU at $\sim$1 kHz nominal. The schematic and four-layer PCB layout were authored in KiCad. Component selection privileges deterministic timing and proprioceptive bandwidth over wireless flexibility: the STM32G431 was chosen over an ESP32 for crystal-less USB CDC and predictable interrupt latency, and the TCA9548A multiplexer keeps four Hall sensors and two ToF sensors on a single I^{2}C bus.

C. Hall sensing—angle and squeeze force

Each MLX90393 reads three axes of magnetic flux from its joint magnet. The B_x and B_y channels give continuous joint angle via $\mathrm{atan2}$ with no wrap discontinuity, provided the policy consumes the raw (B_x, B_y) pair rather than a precomputed angle. The B_z channel is the project's load-bearing sensing contribution: as the fingertip cap is loaded, the PIP slit compresses, the magnet translates axially toward the sensor, and B_z changes monotonically with applied force. The policy receives all three raw channels per sensor and learns the angle-and-force decomposition end-to-end; this avoids both $\mathrm{atan2}$'s branch cut and the need for force calibration in physical units, which an imitation-learning policy does not require to learn the underlying correlations from demonstration.

To validate the design beyond intent, we ran a controlled scale-press experiment: 20 ramped + 20 random-angle presses per assembly across the 300–1300 g range, on both an Index assembly and the 3F assembly. The 3F PIP joint gives a per-press linear fit of $\mathbf{+1.29}$ **counts/g** with $\mathbf{R^2 = 0.64}$ between B_z residual (after subtracting the angle-dependent baseline $B_z^{\mathrm{rest}}(\theta)$ recovered from a separate rotation sweep) and applied scale force. The Index MCP joint, by contrast, shows a flat B_z residual under load—which is consistent with the mechanical design, because the V3 MCP linkage is too stiff to translate the magnet axially.

This is a narrowed claim relative to the proposal: V3 demonstrates force-correlated B_z sensing on *compliant-linkage PIP joints*; rigid MCP joints will require the V4 air-gap revision before the same calibration can be made there. Within that scope, B_z is the differentiating sensing channel of the project—it is what this form-factor reimplementation adds beyond Sunday's public materials as a measurable signal.

V. Pose Tracking and Validation

The production pose-tracking stack is HLoc with a pose-graph batch smoother and a final Gaussian filter at σ=5 frames, validated against AprilGrid ground truth at the camera physical noise floor. This section summarizes the prior backends and the failure modes that drove each pivot (Sec. V-A), the end-to-end evidence supporting the largest pivot (Sec. V-B), the investigation that selected the final smoother (Sec. V-C), and the controlled validation experiment (Sec. V-D).

A. The pivot chain

Table I summarizes the evaluated pose backends and the measured failure mode that motivated each transition. The key pattern is that failures were structural rather than parameter-tuning issues: COLMAP's scale ambiguity required metric anchoring, OKVIS2's finger occlusion required a different camera geometry, the custom GTSAM stack lacked full bundle adjustment, and ORB-SLAM3 hit a reproduction ceiling in map loading. Across these changes, the downstream interface stayed fixed: every backend was scored against the same `parent_map` + tag-frame conventions, the same per-demo quality verdict, and the same `episode.parquet` schema. This common evaluation layer made the AprilGrid validation experiment in Sec. V-D comparable across backends.

B. The pivot evidence: end-to-end ORB-SLAM3 outcome

On identical ORB-SLAM3 settings (same Docker image, same KannalaBrandt8 fisheye intrinsics class, same ArUco initialization), a map built from `session_2026-04-16_085612` (OV9782, V3 glove) achieved **10.9% tracking** with 27 active-map resets across a 68.6 s clip, and **0 of 317 BoW relocalizations** succeeded on a cross-session demo. The same pipeline on Hero 10 + MaxLens hit **99.6% tracking on the map and 99.5% on the localized demo**—a 9× tracking ratio that justifies the camera pivot end-to-end.

C. Trajectory-quality smoother selection

HLoc's per-frame PnP solves are temporally independent; raw frame-to-frame trajectories are visibly spikey. The production trajectory pipeline runs a pose-graph batch state estimator on the raw PnP poses—constant-velocity motion prior, per-frame measurement covariance derived from PnP inlier counts, manifold-aware rotation averaging via windowed quaternion mean—followed by a final temporal filter on each pose axis to clean residual high-frequency noise. We compared candidate temporal filters against the production trajectory-quality gates—a 0.25 s local-straightness threshold $\geq$ 0.80 and a high-frequency RMS threshold $<$ 1.00 mm, calibrated to an ORB-SLAM3 reference demo. A 1-D Gaussian filter at σ=5 frames ($\sim$80 ms / $\sim$2 Hz cutoff) was the simplest method that met both gates, taking three smoke-test demos from straightness 0.73–0.84 to 0.92–0.97 and high-frequency RMS from 1.9–4.4 mm to 0.87–1.24 mm. Mask-only-on-query was rejected (drove straightness from 0.78 to 0.62); GTSAM IMU fusion was rejected because the GPMF IMU's per-frame integration noise dominates the constant-velocity prior at slow hand motion; pose-only bundle adjustment with fixed map points reduces analytically to per-frame PnP, which we confirmed empirically ($\|\text{result} - \text{raw HLoc}\|_\infty \leq 0.04$ mm); geometric-conditioning grid balancing was tested and slightly worsened the metric on 2 of 3 demos; and DROID-SLAM as a relative-motion candidate gave Sim(3) (3D similarity transform: rotation, translation, uniform scale) residuals of 342–756 mm—locally smooth but globally distorted.

D. AprilGrid noise-floor proof

To bound how much further improvement was possible, we built a controlled validation harness using a Kalibr 6×6 AprilGrid (`tag36h11`, 25.796875 mm tags, 0.3 spacing) as truth. Six clips were recorded with the board visible: one mapping clip with the board fully in frame, plus five validation clips spanning static no-cup, static cup-in-hand, and slow board-circle scenarios. Per-frame ground-truth GoPro pose came from `cv2.solvePnP(SOLVEPNP_IPPE_SQUARE)`. We built a SfM map from the mapping clip—extending `glove.process build-model` with a `--skip-tag` flag so the tag could be recovered separately for the truth path—localized each validation clip against it via the production HLoc + σ=5 stack, and Sim(3)-aligned each HLoc trajectory to the AprilGrid truth on shared frames. Two results bound everything that comes after.

Static no-cup: HLoc 8.3 mm p95 vs. AprilGrid 8.1 mm p95. HLoc's $p95$ error is within **0.2 mm** of the AprilGrid PnP $p95$ noise floor on the same frames.

Static cup-in-hand: HLoc 29.2 mm p95 vs. AprilGrid 31.7 mm p95. HLoc is *slightly better* than the gold-standard board-PnP because the σ=5 smoothing helps; both struggle by the same magnitude when the cup occludes view.

The wobble in production trajectories on the validated motion regimes is the camera's intrinsic measurement noise plus cup-occlusion physics, not an algorithmic deficiency. Further improvement on these regimes requires capture-side hardware or scene changes—a higher-resolution camera, a denser-textured workspace background, more SfM coverage of demo viewpoints—rather than further work on the SLAM stage. The fast-motion regime was not separately bounded; see Sec. VIII.

VI. Time and Spatial Sync

The capture stack writes two independent streams: the GoPro Hero 10 records video at 60 fps to its own SD card, and a continuous master-tape recorder on the host writes proprio (Hall + ToF at 30 Hz) and IMU ($\sim$1 kHz) to a per-day directory. The two streams must align to within a video frame ($\sim$17 ms) at the moment the data are sliced into per-demo session directories during ingest.

We use GoPro Labs firmware combined with GoPro's UTC precision-time QR page [13] for the absolute clock. The Labs firmware reads the animated QR directly into the camera RTC, so MP4 timecodes are millisecond-accurate UTC from the moment the camera starts recording. This replaces UMI's `calibrate_timecode_offset.py` approach, which decodes a laptop-rendered QR from the video itself: that approach works but requires a QR visible at the start of every recording. GoPro Labs sync is once per session (or per week), then just record.

One systematic offset remains. The GoPro Hero 10's encoder pipeline stamps timecodes $\sim$333 ms *after* the actual shutter moment; we measured this empirically by scrubbing a Hall-sensor spike against the visible finger bend that produced it, and compensate at ingest with a constant offset.

TABLE I
THE EIGHT POSE-TRACKING BACKENDS, IN CHRONOLOGICAL ORDER. BOLD ROW MARKS THE CURRENT PRODUCTION STACK.

#	Backend	Active	What it did differently	Why it ended	What carried forward
1	DROID-SLAM [11] (1st try)	Feb 17	Learned dense optical-flow SLAM from monocular video (visual only, no inertial)	Lietorch CUDA timeout + GPU SIFT crash on Colab; cloud-only stack with insufficient debug latitude	Production stages should avoid uncached cloud-GPU dependencies
2	COLMAP SfM	Feb 18–21	Pure offline SfM across 17 demo sessions; metric scale recovered post-hoc from IMU integration	Cross-session scale CV **25.4%**, position RMSE **18 cm**, **0.5%** of frames at physically impossible velocities	Pivoted to stereo for metric anchoring at acquisition, not post-hoc
3	Custom stereo SIFT triangulation	Feb 21–Mar 1	In-house stereo geometry; SIFT cross-matches, triangulate, integrate	Ruler test 1.2 m → 1.6 m (33% scale error)	Triangulation alone is insufficient; need full VIO that fuses IMU and visual constraints
4	OKVIS2-X stereo [12] (OV9782)	Mar 1–Apr 8	Production-grade stereo-inertial VIO; Kalibr 0.81 px reprojection; BNO085 IMU	Worked pre-assembly (1.28 m on 1.2 m ruler, 6.7% error). Killed by **finger occlusion** post-glove-assembly: 81° HFOV stereo overlap collapsed from 36° clean to −13°	Failure was geometric, not algorithmic; CIL237 fisheye lenses tried but ultimately pivoted past stereo entirely
5	Custom GTSAM v1/v2/v3	Apr 1–7	In-house factor graphs to replace OKVIS2; CoTracker for keypoints + GTSAM for optimization	**v2 worked on a single hand-picked session: 0.376 m vs. 0.380 m ground truth (0.5% scale).** On novel manipulation sessions, scale collapsed by 21× because absence of full BA meant landmarks were frozen at birth	Full bundle adjustment is required for cross-session generalization; archived
6	ORB-SLAM3 + UMI Hero 10	Apr 18–24	Switched cameras to GoPro Hero 10 + MaxLens Mod 1.0 (155° FOV); reused UMI's published intrinsics + Docker; SLAM via UMI's `chicheng/orb_slam3`	First mapping run: **99.6% tracking on 83 s map clip, 99.5% on the localized demo, demo-start 1.2 cm from mapping endpoint**. Hit ∼20% community reproduction ceiling; `chicheng`'s source not published	Validated the production stack and the Hero 10 + MaxLens hardware path
7	**HLoc + smoother + Gaussian** σ=5	Apr 24–present	Per-frame visual localization: SuperPoint + LightGlue + NetVLAD + COLMAP via HLoc; pose-graph LM smoother with CV Huber prior; final 1-D Gaussian filter at σ=5 frames (∼2 Hz cutoff at 60 fps)	**Current production.** Smoke test on three demos that ORB-SLAM3 had failed or diverged: 100% localized + `auto_pass` on all three	Same two-stage architecture as ORB-SLAM3; learned features + retrieval give materially higher robustness on hard scenes
8	DROID-SLAM revisited	Apr 25	Tested as relative-motion candidate for hybrid HLoc-anchor + DROID-relative factor graph; patched `terminate()` to return dense per-frame poses	Competitive but not clearly better than HLoc + σ=5; Sim(3) residuals 342–756 mm median (locally smooth, globally distorted)	Hybrid HLoc + DROID factor graph deferred

VII. POLICY TRAINING

A. Architecture

We use ACT [2], the action-chunked transformer policy from ALOHA, in preference to a pure diffusion policy [9]. The encoder is a CVAE over the action chunk (4 layers); the decoder is DETR-style (7 layers) and conditions on a frozen DINOv2 ViT-S/14 visual backbone. The observation vector is 23-dimensional (relative end-effector position 3 + 6D rotation 6 + raw Hall channels 12 + ToF 2); the action vector is 21-dimensional (same structure, no ToF). Hall channels enter the policy as raw (B_x, B_y, B_z) per sensor rather than precomputed angles—both to avoid the atan2 branch cut and so that B_z force information is preserved without a calibration step. Total parameters: 38.7 M, of which 16.6 M are trainable. Action horizon is 300 frames (5 s at 60 fps), chosen because the mug-on-coaster task routinely has the coaster out of frame mid-pickup; long-chunk planning substitutes for the temporal memory that ACT's two-frame observation horizon does not provide.

B. Dataset

The training set is 182 mug-on-coaster demonstrations recorded across four visually distinct environments (Figure 5), with 155 used for training and 27 reserved for validation (`val_fraction = 0.15`, `val_seed = 42`). Each demonstration averages ∼500 GoPro frames at 60 fps; total ∼77,000 frame samples per epoch. The set is what remained after the label-quality audit in Sec. VII-D excluded 40 of the original 172 sessions for silent SLAM re-anchoring artifacts, plus 50 additional clean demonstrations recorded the following day.

C. Training runs

Final training used: `kl_weight` = 50, batch 32, learning rate 1×10^{-4} (cosine schedule), seed 42, action horizon 300 frames (5 s at 60 fps), augmentation on, 150-epoch budget. Best checkpoint at epoch 38. Five recipe ablations on the same baseline (reduced `kl_weight`, endpoint and ramped position-loss weighting, free-bits CVAE-collapse fix, per-frame SE(3) delta action targets, and inference-time temporal action

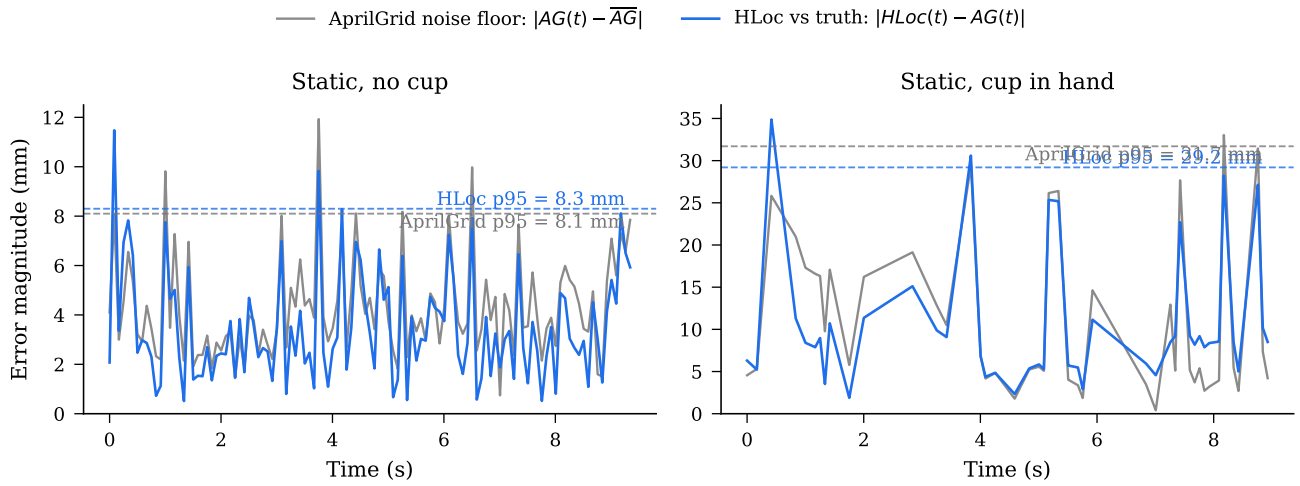


Fig. 4. AprilGrid validation: per-frame error magnitude over time. **Blue**: HLoc-vs-truth disagreement, $|\mathrm{HLoc}(t) - \mathrm{AprilGrid}(t)|$ after Sim(3) alignment on shared frames. **Gray**: AprilGrid noise floor on the truth side, $|\mathrm{AprilGrid}(t) - \overline{\mathrm{AprilGrid}}|$. **(left)** Static no-cup: blue tracks gray almost exactly; HLoc-vs-truth p95 = 8.3 mm sits 0.2 mm above the AprilGrid noise floor of 8.1 mm. **(right)** Static cup-in-hand: cup occlusion drives both error sources up by ∼4×; HLoc remains at the noise floor (29.2 mm vs. 31.7 mm).

Fig. 5. Environment diversity in the mug-on-coaster training set: four visually distinct workspace settings. Scene variety is what the policy sees during training; only mug-on-coaster is shown because only mug-on-coaster was used to train the policy reported here.

ensembling) returned to the same plateau without changing the final design.

For interpretation, the AprilGrid validation in Sec. V-D bounded HLoc's measurement noise at 8.3 mm $p95$ on static frames; the policy's 28.8 mm position L_1 is therefore roughly $3.5\times$ the truth-source's own measurement floor, with the residual dominated by model imperfection rather than label noise. This is an offline metric against the production trajectory ground truth; deployed task-success rate is not measured here—an actuated arm and a held-out novel scene would be required to convert it into a task-completion claim.

TABLE II
FINAL ACT POLICY RESULTS ON THE HELD-OUT VALIDATION SET, AGAINST THE PRODUCTION HLOC + σ=5 GROUND TRUTH.

Dataset	182 mug-on-coaster demonstrations
Train / val split	155 / 27 sessions (`val_seed` = 42)
Label source	HLoc + σ=5 Gaussian smoothing
Best epoch	38
Position L_1	**28.8 mm**
Rotation L_1	**6.31°**
Hall L_1	**736 LSB**

D. Trajectory-Quality Filtering

A per-demo trajectory-smoothness audit across the 172 candidate mug-coaster sessions found that **40 of 172 (23%) contained at least one unphysical single-frame jump** > 50 **mm** (3 m/s at 60 fps), with two worst demos showing 340 mm teleports. These were silent SLAM pose-graph re-anchoring events not flagged by the `is_lost` bit. The production trajectory-quality gate combines tracking percentage with the local-straightness and high-frequency-RMS thresholds in Sec. V-C; the 40 demos failing the gate were excluded from training.

VIII. Discussion and Limitations

Honest limitations.

- Hall force sensing is validated only on compliant-linkage PIP joints. Rigid MCP joints will require the V4 air-gap revision before they can be calibrated similarly. The force-sensing claim is correctly narrowed to PIP.
- The reported 28.8 mm position L_1 is an offline metric against trajectory ground truth, not a deployed task-completion rate. Single-task scope further limits cross-task generalization claims.

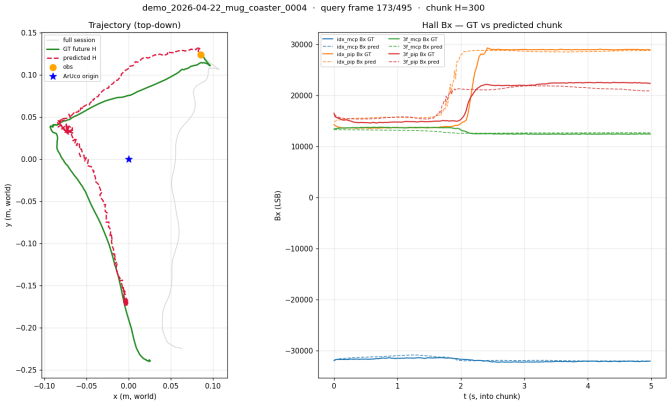


Fig. 6. Action prediction vs. ground truth on the held-out demo `demo_2026-04-22_mug_coaster_0004` (production checkpoint, chunk horizon $H = 300$). **Left**: top-down workspace trajectory in the gravity-aligned tag frame—the full session in light gray, the ground-truth future chunk (5 s of motion) in green, and the model's predicted chunk in red dashed. **Right**: the four Hall sensors' B_x channels over the same chunk, ground truth solid and prediction dashed; the predicted Hall trace tracks the squeeze cycle through the grasp event.

- All 182 demonstrations were recorded by a single operator. Hand kinematics, demonstration style, and Hall-magnet seating are constants in this dataset; cross-wearer generalization has not been evaluated.
- HLoc localization requires a per-environment SfM map. A new scene needs a fresh mapping clip and a rebuilt atlas before demos in that scene can be aligned (Sec. V). The pipeline is therefore not zero-shot to novel locations.
- The AprilGrid noise-floor claim in Sec. V-D was bounded on static-no-cup and static-cup-in-hand frames plus a slow board-circle scenario. Fast pickup-and-place motions were not separately bounded against ground truth, so the 8.3 mm $p95$ figure does not transfer directly to high-velocity frames.

Future work. A V4 hardware revision is in early planning, with three targets informed by the analyses in this report. First, revised joint geometry that extends Hall force calibration from compliant PIP to rigid MCP joints (Sec. IV-C). Second, a hardware sync trigger that eliminates the constant encoder-pipeline delay correction (Sec. VI) and provides per-frame spatial sync. Third, on-glove stereo cameras revisited with the finger-occlusion lessons from the OKVIS2 era baked into lens choice and image masking (Table I, row 4). Capture-side improvements suggested by the AprilGrid bound (Sec. V-D)—higher-resolution sensors, denser workspace texture, and more SfM coverage of demo viewpoints—would lower the physical noise floor that currently bounds pose accuracy. Beyond hardware, actuated-arm deployment of the trained policy would convert the offline L_1 metric into a task-completion rate; bimanual capture and cross-task transfer extend the dataset along orthogonal axes.

IX. Conclusion

We presented a four-month solo reimplementation of Sunday Robotics' published Skill Capture Glove form factor that adds the missing measurement layer: a Hall-instrumented glove and custom DAQ PCB whose primary sensing contribution is a force-correlated B_z signal on compliant PIP joints; a video-and-proprioception capture pipeline synchronized via GoPro Labs UTC; an HLoc + σ=5 SLAM stack validated to within 0.2 mm of the AprilGrid PnP noise floor on static frames; and an ACT policy trained on 182 demonstrations reaching 28.8 mm position L_1 on a held-out validation set. The system emits training-ready episode files—one per demonstration, with a stable schema—for downstream imitation-learning use. Code, schema, and a reproducible CLI for the full pipeline are available at https://github.com/corneliusgruss/skill-capture-glove.

References

[1] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song, "Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots," in *Proceedings of Robotics: Science and Systems (RSS)*, 2024.

[2] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn, "Learning fine-grained bimanual manipulation with low-cost hardware," in *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[3] Sunday Robotics, "Sunday robotics skill capture glove," 2025, https://sunday.ai/.

[4] P.-E. Sarlin, C. Cadena, R. Siegwart, and M. Dymczyk, "From coarse to fine: Robust hierarchical localization at large scale," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[5] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Superpoint: Self-supervised interest point detection and description," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018.

[6] P. Lindenberger, P.-E. Sarlin, and M. Pollefeys, "Lightglue: Local feature matching at light speed," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[7] R. Arandjelovic, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "Netvlad: Cnn architecture for weakly supervised place recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[8] J. L. Schönberger and J.-M. Frahm, "Structure-from-motion revisited," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[9] C. Chi, S. Feng, Y. Du, Z. Xu, E. Cousineau, B. Burchfiel, and S. Song, "Diffusion policy: Visuomotor policy learning via action diffusion," in *Proceedings of Robotics: Science and Systems (RSS)*, 2023.

[10] C. Campos, R. Elvira, J. J. Gómez Rodríguez, J. M. M. Montiel, and J. D. Tardós, "Orb-slam3: An accurate open-source library for visual, visual-inertial, and multimap slam," *IEEE Transactions on Robotics*, vol. 37, no. 6, pp. 1874–1890, 2021.

[11] Z. Teed and J. Deng, "DROID-SLAM: Deep visual SLAM for monocular, stereo, and RGB-D cameras," in *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

[12] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 314–334, 2015.

[13] GoPro, "Gopro labs: Precision time UTC," 2024, https://gopro.github.io/labs/control/precisiontime_utc/.