How Visible Are Silent Failures in Bimanual Manipulation? An Observability Study of False-Success Detection in Simulated ALOHA Episodes

Author Names Omitted for Anonymous Review

***Abstract*—Imitation-learning policies for bimanual manipulation inherit the quality of the success labels attached to their training episodes, and on bimanual platforms such as ALOHA those labels are usually produced by the robot's own scripted success check. A particularly damaging error is the *false success*: an episode the robot logs as a success when the task outcome was actually wrong. We ask a narrow but practical question about these episodes on two-arm tasks. Once a bimanual episode has already been flagged as a success, how much of the information needed to overturn that label is present in proprioception across the two arms, and how much requires vision? We build a simulated testbed on two bimanual ALOHA tasks, induce failures through environment perturbations rather than label edits, label every episode by privileged simulator state that the detector never sees, and keep only episodes the robot flagged as successful. We then compare detectors restricted to two-arm proprioception against a vision-based detector. We find that recoverability spans a wide range: in cube transfer the false successes are almost fully recoverable from joint data alone, while in peg insertion proprioception recovers only part of them and a vision detector closes most of the gap. We also show that the proprioceptive separability we measure rests on velocity differences far below any realistic sensor noise floor, so it is best read as an optimistic upper bound that a noiseless simulator inflates. The result is directly relevant to anyone curating bimanual demonstration data: the modality that catches a silent failure depends on the task, and clean-simulation proprioception numbers overstate what a real two-arm system would recover. We release the generation and evaluation pipeline.**

I. Introduction

Whole-body and bimanual manipulation policies are increasingly trained by imitation from large demonstration corpora [5, 7, 4, 1], and every one of those corpora rests on the assumption that each episode carries a correct success or failure label. On bimanual platforms such as ALOHA [8], the label is generated automatically, by the robot's own success detector or a scripted end-of-trajectory check. When the check fires on an episode that did not actually accomplish the task, the resulting *false success* enters the training set as a positive example, and the policy has no way to know that the two-arm behavior it is imitating ended badly. As humanoid and bimanual systems scale their data collection, the cost of these silent labeling errors scales with them.

False successes are harder to find than ordinary execution failures. A stalled joint or a violent jerk leaves an obvious trace in the sensed trajectory; a cube placed two centimeters off target, or a peg that looks seated but is not, may leave almost none. The practical question for anyone trying to clean a bimanual training set is therefore not whether a failure occurred, but which sensing channel carries enough signal to recover it after the robot has already declared success. This is a question about *observability*: given the proprioceptive and visual channels available on a two-arm platform, which failures are in principle recoverable from which channel?

This paper studies that question directly in simulation, on two bimanual ALOHA tasks. We deliberately avoid the shortcut of injecting failures as label edits, because a detector trained on such data learns to recover the injection rule rather than a real failure signature. We instead run a scripted bimanual controller in a physics simulator, perturb the environment so that genuine misses occur, and read the true outcome from privileged simulator state that the detector is never given. We keep only the episodes the robot flagged as successful, so that the task is exactly the one a curation pipeline faces: separate the real successes from the false ones using only observable signals.

Our contribution is an observability analysis rather than a new detection method. We report, for two bimanual manipulation tasks, how far a proprioception-only detector and a vision detector get on the false-success problem, and we are explicit about where the proprioceptive signal comes from and how fragile it is. Two findings stand out: First, recoverability is not uniform: transfer failures are almost fully visible to joint data, while insertion failures are only partly visible and need vision to be caught reliably. Second, the proprioceptive separability we measure is driven by velocity differences on the order of one part in a thousand, which a noiseless simulator resolves perfectly but a real robot's sensors would not. We therefore treat the proprioception numbers as an upper bound. The pipeline is released so the analysis can be extended to other bimanual tasks, policies, and modalities.

II. Related Work

Bimanual manipulation datasets. Large demonstration corpora have driven recent progress in robot learning. Open X-Embodiment [5] aggregates data across many robots and labs, BridgeData V2 [7] and DROID [4] provide large single-embodiment collections, and LeRobot [1] standardizes the

data format and supplies widely used reference datasets. The ACT line of work [8] established the low-cost bimanual ALOHA platform that our testbed builds on. These efforts target scale and coverage and do not provide a labeled measure of annotation quality.

Failure and anomaly detection. Detecting execution failures from proprioception and force is well studied, with residual-monitoring and learned sequence-model approaches [6, 3]. These methods work precisely because the failures they target leave a signature in the sensed trajectory: a collision spikes the measured force, a stalled actuator flattens a velocity profile, a dropped object changes the load the arm carries. The case we study is the opposite one, where the trajectory looks nominal but the outcome is wrong, and it has received much less attention. On bimanual platforms the situation is further complicated. The two arms must coordinate, and a failure in one arm can be masked or compensated by motion of the other, so the question of which sensing channel carries the failure signal is not inherited from the single-arm case but has to be measured on the two-arm system directly. Our study is a first step in that direction: rather than proposing a detector, we ask how much of the false-success signal is in principle available to each channel on two representative bimanual tasks.

Success detection and reward models. Vision-based success classifiers and learned reward models judge task outcome from images [2], and are the natural tool for catching false successes. What has been missing is a controlled measurement of how much of the false-success signal lives in proprioception versus vision once a bimanual robot has already reported success.

III. Testbed Design

Both tasks derive from the bimanual ALOHA simulator [1, 8]: a cube transfer task, in which one arm hands a cube to the other, and a peg insertion task, in which the two arms coordinate to seat a peg. We chose these two because they sit at opposite ends of the coordination spectrum that bimanual systems care about. Transfer is a loosely coupled handoff where each arm can be reasoned about almost independently, while insertion is a tightly coupled task where the two arms must agree on a shared target pose for the task to succeed. If the sensing channel that carries a failure signal were the same for both, the distinction would not matter; part of what we report is that it is not. We use the scripted end-effector controllers that ship with the simulator so that nominal two-arm behavior is competent and reproducible, and we draw the true outcome from the simulator's privileged object state.

Inducing failures. We do not edit labels. For each episode we apply randomized environment perturbations (Table I): a small planar jitter and yaw perturbation of the manipulated object, and randomized object friction. The scripted controller then runs to completion. Some perturbed episodes still succeed and some do not; the failures are a consequence of the physics, not of an annotation.

TABLE I
Testbed composition (flagged-success episodes only, seed 42).

	Transfer	Insertion
Flagged-success episodes	500	500
True successes	262	341
False successes	238	159
False-success rate	47.6%	31.8%
Object planar jitter	1.5 cm	0.5 cm
Object yaw jitter	15°	5°
Object friction range	0.15–0.9	0.15–0.9
Added observation noise	none	none

Two labels, kept apart. Each episode carries two distinct labels. The *robot flag* is a cheap success heuristic computed from proprioception only, standing in for the telemetry-based check a real bimanual robot would use. The *ground-truth outcome* is read from privileged simulator state, the object pose relative to its target, which the detector never observes. A false success is an episode where the robot flag fired but the ground-truth outcome is a failure. We retain only flagged-success episodes, so the dataset is exactly the pool a curation step would inspect.

Failure modes. Modes are assigned after the fact from the ground-truth state, not injected. In transfer the failures are dominated by a misplaced cube, with a small number of drops; in insertion the single observed mode is a peg that ends up not seated near the socket. We report whichever modes occur rather than forcing a fixed taxonomy.

IV. Evaluation Protocol

The task is stated as follows: given a bimanual episode already flagged as a success, predict whether it actually succeeded. The headline metric is *false-success recall*, the fraction of truly failed episodes that the detector flags, because that is the quantity a curation step cares about and because overall accuracy is misleading when one class dominates.

We compare two detectors. **Baseline A (proprioception)** uses summary statistics of the twelve arm joints' velocity over the full episode (six per arm across the two arms): three global statistics (the root-mean-square, peak, and standard deviation of the per-step joint-velocity magnitude) together with the root-mean-square, peak, and standard deviation of each individual joint's velocity, for thirty-nine features in total. No privileged state, perturbation parameter, or failure-mode label is used as a feature. **Baseline C (vision)** uses three features read from the rendered camera image of the final state: the normalized image centroid (c_x, c_y) and the pixel area of the manipulated object, obtained by color-thresholding the rendered frame. These come from the image alone; the detector never reads object pose from simulator state. Both detectors are gradient-boosted tree classifiers with class balancing, trained on a fixed split and evaluated on a held-out test set of 150 episodes. The seed is fixed at 42.

As a diagnostic that the proprioceptive features are not trivially separating the classes, we report Cohen's d between true

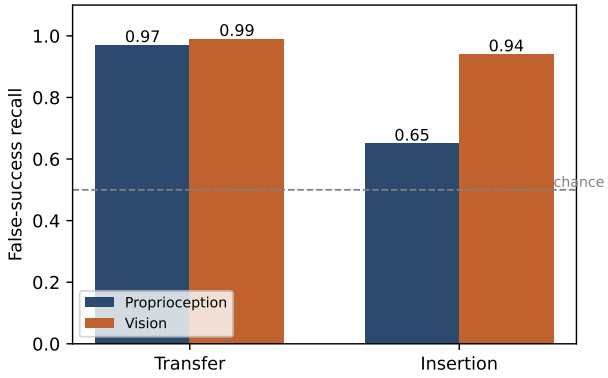

Fig. 1. False-success recall by sensing modality. Transfer failures are almost fully recoverable from two-arm proprioception; insertion failures need vision to close the gap.

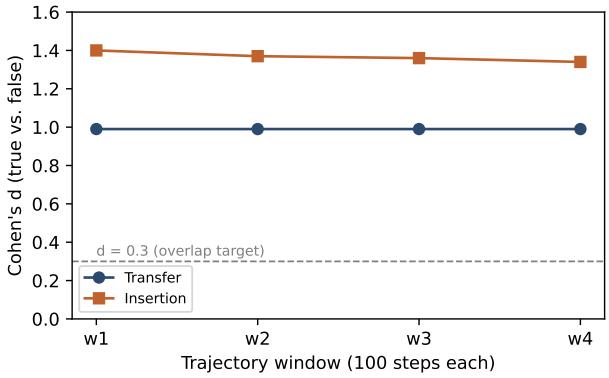

Fig. 2. Per-window Cohen's d between true and false successes. The separating signal is present across the whole trajectory in both tasks, well above the overlap target.

TABLE II
DETECTOR METRICS ON THE HELD-OUT TEST SET (150 EPISODES).

Task	Detector	Acc.	FS rec.	TS rec.	F1
Transfer	Proprio.	0.973	0.972	0.975	0.973
Transfer	Vision	0.987	0.986	0.987	0.987
Insertion	Proprio.	0.760	0.646	0.814	0.727
Insertion	Vision	0.973	0.938	0.990	0.969

TABLE III
PER-WINDOW SEPARABILITY VS. ABSOLUTE MEAN DIFFERENCE (TRANSFER).

Window	mean (true)	mean (false)	Cohen's d
t000–100	0.8598	0.8608	0.987
t100–200	0.2002	0.1929	0.993

and false successes for every feature and for each trajectory window. Large d indicates that the failure is, in fact, visible to proprioception.

V. RESULTS

Recoverability differs sharply by task. Fig. 1 and Table II give the headline numbers. In transfer, proprioception alone recovers 97% of the false successes and vision adds little, because the failures change the load the arm carries and so are written plainly into the joint trajectory. In insertion, proprioception recovers only 65%, while the vision detector reaches 94%. Insertion is where the modality gap is real.

The proprioceptive signal is present in every window. Fig. 2 shows Cohen's d between true and false successes within each 100-step window. For transfer it sits near 1.0 throughout, and for insertion above 1.3 in every window. There is no segment of the trajectory in which the two classes become indistinguishable, which rules out the possibility that a particular feature window is doing the separating. The signal is concentrated in a few joints rather than spread evenly across the two arms.

But the signal is tiny in absolute terms. Table III pairs each window's Cohen's d with the actual difference between the class means. In transfer's first window the mean feature differs by less than 0.001 in normalized units, yet $d \approx 0.99$. The large effect size comes almost entirely from near-zero within-class variance, which the deterministic simulator produces and a real sensor would not. The proprioceptive separability is therefore real in this simulator and fragile outside it.

VI. DISCUSSION AND LIMITATIONS

The headline of this study is a caution as much as a result. A clean reading of the numbers would say that false successes are mostly recoverable from proprioception, but that reading would be wrong for two reasons, and stating them plainly is the point of the paper.

Transfer is essentially physical-failure detection. The transfer failures we induced change the mass the arm carries, so they show up in the joint trajectory, and a proprioception detector catches them at ceiling. These are not the silent failures the problem is about; they are ordinary execution failures that happen to be mislabeled as successes by a loose flag. The only task here that exhibits a genuine proprioception versus vision gap is insertion.

The proprioceptive separability is a simulation artifact. The velocity differences that separate the classes are on the order of one part in a thousand. We added no observation noise. A real robot's encoder and velocity estimates carry noise that would bury a signal this small, so the 0.97 and 0.65 numbers should be read as optimistic upper bounds rather than as expected field performance. This makes the case for vision stronger, not weaker.

The insertion signal is partly a policy artifact. The scripted controller steers toward the true peg pose, so when the peg is jittered the arm's trajectory adapts, and that adaptation correlates with eventual success. A learned, closed-loop, or fixed open-loop bimanual policy would distribute this signal differently. The insertion proprioception number is therefore tied to the oracle controller and should not be read as a property of the task alone.

Implications for bimanual data curation. The practical reason to care about any of this is that bimanual demonstration sets are being collected at scale, and the success labels in them are largely trusted without audit. Our results suggest two concrete things for anyone building such a pipeline. The first is that a proprioception-only audit is not safe to rely on for the failures that matter most. The transfer result looks encouraging in isolation, but it is encouraging for the wrong reason: those failures perturb the arm's own dynamics and would be caught by any competent execution monitor, label or no label. The failures that slip through a loose success check and into the training set are the quiet ones, and on insertion those are exactly the ones proprioception misses. The second is that the value of a vision channel is task-dependent and worth measuring before committing to it. On transfer, adding vision buys almost nothing; on insertion it closes most of the gap. A team with a fixed labeling budget would do better to spend it on visual verification of the tightly coupled tasks than to spread it uniformly. Neither conclusion is specific to ALOHA, but both are easiest to establish on a platform like it where privileged state is available to define ground truth.

Scope. The study covers two simulated bimanual tasks, a single random seed, 500 flagged-success episodes per task, and an oracle scripted policy. The failures that arise are dominated by one or two modes per task rather than a broad taxonomy. We report a force-augmented detector in the released code but omit it here because it did not differ measurably from the proprioception detector. The natural next steps are multiple seeds with confidence intervals, learned bimanual policies in place of the oracle controller, calibrated proprioceptive sensor noise so that detectability reflects realistic conditions, and real-robot two-arm episodes.

VII. Conclusion

We studied how recoverable false-success failures are from different sensing modalities on two bimanual manipulation tasks, using a simulated ALOHA testbed in which failures arise from physics rather than label edits and ground truth is hidden from the detector. Recoverability ranges from near-complete for transfer, where the failures perturb the arm's own dynamics, to partial for insertion, where vision is needed to close the gap. We further show that the proprioceptive separability we measure rests on sub-sensor-noise velocity differences and on the oracle controller, so it overstates what a real system would recover. The honest takeaway for bimanual training-data curation is that catching silent manipulation failures generally requires exteroception, and that clean-simulation proprioception results on this problem should be treated with care.

Data and Code Availability

The episode-generation and evaluation pipeline, including the feature extractors and the summary used to produce all figures and tables in this paper, will be released publicly upon acceptance. (Repository link omitted for anonymous review.)

References

[1] Remi Cadene et al. LeRobot: State-of-the-art machine learning for real-world robotics in PyTorch. https://github.com/huggingface/lerobot, 2024.

[2] Yilun Du et al. Vision-language models as success detectors. arXiv:2303.07280, 2023.

[3] Arda Inceoglu et al. FINO-Net: A deep multimodal sensor fusion framework for manipulation failure detection. In *Proc. IEEE/RSJ IROS*, 2021.

[4] Alexander Khazatsky et al. DROID: A large-scale in-the-wild robot manipulation dataset. In *Proc. RSS*, 2024.

[5] Open X-Embodiment Collaboration. Open X-Embodiment: Robotic learning datasets and RT-X models. arXiv:2310.08864, 2023.

[6] Daehyung Park, Yuuna Hoshi, and Charles C. Kemp. A multimodal anomaly detector for robot-assisted feeding using an LSTM-based variational autoencoder. *IEEE RA-L*, 3(3):1544–1551, 2018.

[7] Homer Walke et al. BridgeData V2: A dataset for robot learning at scale. In *Proc. CoRL*, 2023.

[8] Tony Z. Zhao, Vikash Kumar, Sergey Levine, and Chelsea Finn. Learning fine-grained bimanual manipulation with low-cost hardware. In *Proc. RSS*, 2023.