

ISR vs uniform downsampling vs full data — closed-loop success on human teleop demos (ALOHA transfer-cube)

Experiment 2 · 3 arms × 3 seeds × 200 rollouts · 2026-09-14

1. Question

The ISR paper (Yang et al., IROS 2026, arXiv:2606.22907) claims that resampling demonstrations uniformly in *information* (motion + acceleration) instead of *time* raises policy success by ~24 pt over 3× time-uniform downsampling, at the same $\frac{1}{3}$ frame budget, on real robots. It never compared against the full, untouched data, never ran in simulation, and never used joint space.

We test, on public human teleop data with a fully reproducible closed-loop evaluation:

1. **ISR vs uniform 3×** at equal frame budget — does *which* frames matter?
2. **ISR vs raw** — does compression cost anything against using every frame?
3. **Playback rate** — what happens when a policy trained on compressed frames is executed at the native control rate?

2. Data

lerobot/aloha_sim_transfer_cube_human: 50 episodes, a human teleoperating two ALOHA arms in MuJoCo, 400 frames per episode at 50 Hz, 14 joint positions, one 480×640 top camera. Task: right arm picks the cube and hands it to the left arm.

The demos carry real operator timing variance:

metric	value
speed variability (std / mean)	1.01
stop-go segments per 8 s episode	4.9
idle frames per episode	20 % (range 7–34 %)
handover time across episodes	3.6 s $\pm$ 0.7 s (2.3–6.2 s)
high-jerk frames	19.5 %

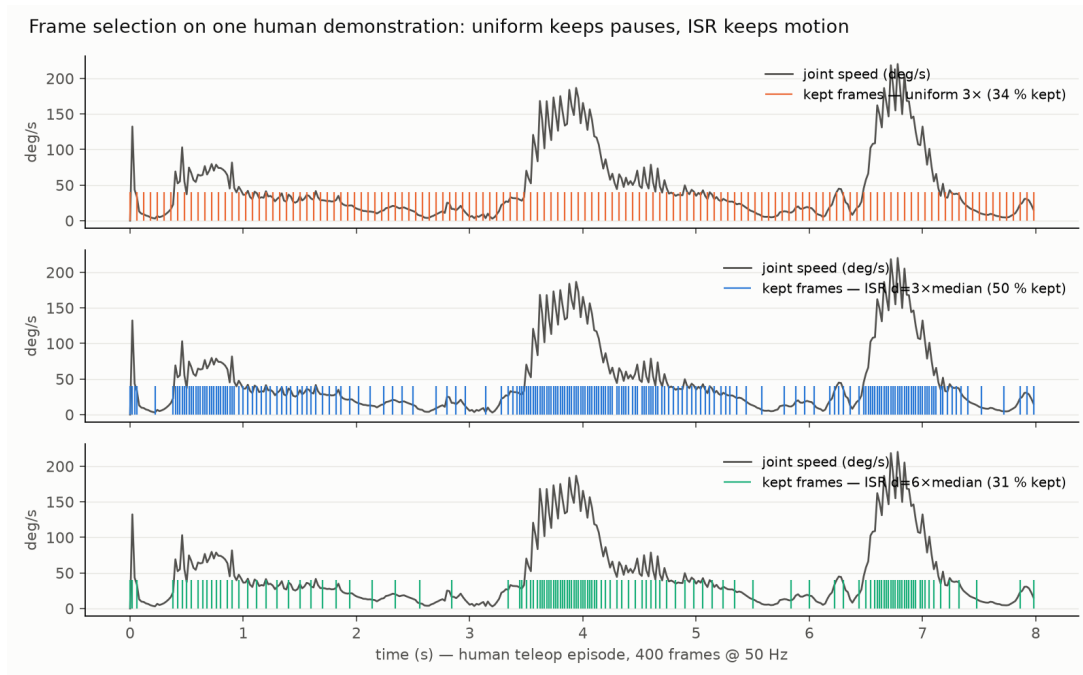
3. ISR as applied

Joint space, 12 arm joints in **degrees** (the paper’s numbers are end-effector meters and do not transfer; applied literally they keep 93 % of frames and, with the acceleration term at 88 % of the information, degenerate to uniform sampling).

Knobs were chosen from a full lookup table (tables/transfer_cube/grid.md: $d_target \times \lambda_acc \rightarrow$ kept % and pause-frames kept %), following the paper’s protocol:

knob	value	rationale
frame budget	$\frac{1}{3}$ (32 % kept)	same as the paper’s baseline
d_target	3.5°	solves for $\frac{1}{3}$ budget
λ_acc	0.0004	acceleration $\approx$ 20 % of total information
gripper force-keep	2.84° (5 % of range)	paper’s rule

Effect on the data: 20,000 $\rightarrow$ 6,412 frames; **pause frames kept 4 %** (uniform 3 $\times$ keeps 34 %); grasp/handover frames force-kept.

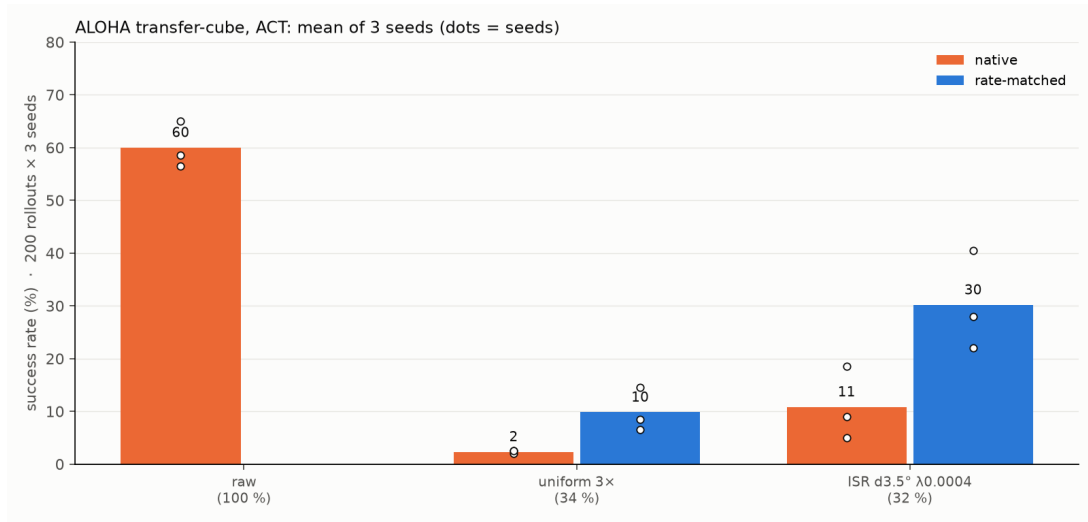


4. Arms and protocol

arm	rule	frames	kept
raw	all frames	20,000	100 %
uni3	every 3rd frame (paper baseline)	6,700	34 %
isr	ISR, $d\ 3.5^\circ$, $\lambda\ 0.0004$	6,412	32 %

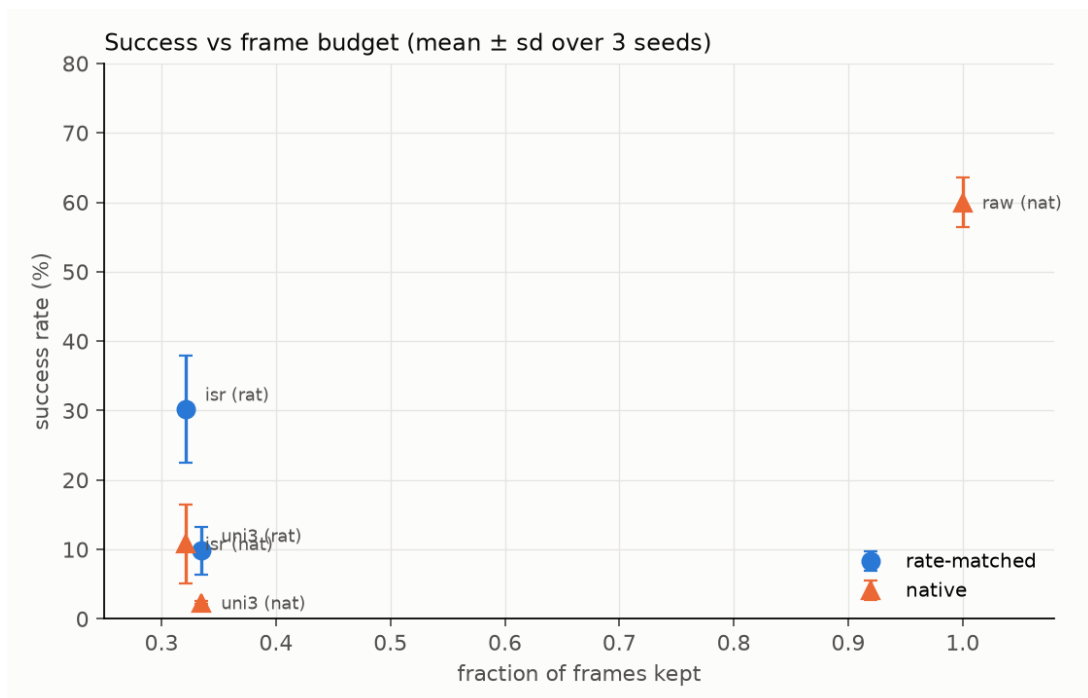
- **Policy:** LeRobot ACT, ALOHA defaults — ResNet-18, transformer dim 512, chunk 100, batch 8, lr $1e-5$, VAE on. **20,000 steps**, seeds 1000/2000/3000. Identical for every arm.
- **Eval:** gym_aloha/AlohaTransferCube-v0, **200 rollouts** per checkpoint, 400-step budget, success = cube in the left gripper.
- **Playback:** *native* — one policy action per sim step (a $\frac{1}{3}$ -budget policy moves 3× faster than the demos); *rate-matched* — each action held 3 sim steps, same 400-step budget.

5. Results



arm	kept	native SR %	rate-matched SR %	per-seed (native)	per-seed (rate-matched)
raw	100 %	60.0 ± 3.6	–	56 / 58 / 65	–
uni3	34 %	2.3 ± 0.2	9.8 ± 3.4	2 / 2 / 2	6 / 14 / 8
isr	32 %	10.8 ± 5.7	30.2 ± 7.7	5 / 18 / 9	22 / 40 / 28

mean ± sd over 3 seeds; each cell 200 rollouts (±7 pt).



6. Findings

1. **At equal $\frac{1}{3}$ budget, ISR beats uniform $3\times$ by $3\times$ — 30 vs 10 % rate-matched, 11 vs 2 % native.** Every seed agrees (22/40/28 vs 6/14/8). Selecting frames by information rather than by the clock is the better way to compress. The paper’s core claim reproduces on public human data in joint space.
2. **Neither compressed arm approaches full data.** Raw reaches 60 %. With only 50 demos, throwing away two-thirds of the frames costs 30 pt even with the best selection. Compression is not free; ISR only halves the damage.
3. **Playback rate is decisive.** Compressed-data policies executed at native rate collapse (2 %, 11 %); holding each action 3 steps recovers 10 % and 30 %. ISR outputs carry no timing, so the deployment side must restore it — the paper’s real-robot loop did this implicitly.
4. **Reconciling with the earlier run** (same task, 100 rollouts, 60k steps): uniform $3\times$ scored 31/20 %, ISR- $\frac{1}{3}$ 42/40 %, raw 62 %. With 20k steps and 200 rollouts the compressed arms are lower in absolute terms, the ranking and the ISR:uniform ratio are unchanged. Compressed arms appear to need more training steps than raw to converge — a further cost of compression.

7. Verdict

- ISR is a **better compressor** than time-uniform downsampling: same budget, $3\times$ the success.
- ISR is **not a substitute for data**: at $\frac{1}{3}$ budget it loses 30 pt to raw on 50 demos. The earlier run showed ISR at 50 % kept can match raw; the knee lies between $\frac{1}{3}$ and $\frac{1}{2}$.
- **Deployment must be rate-aware** or any compressed policy fails.
- Acceleration weighting adds little on human data (earlier $\lambda=0$ ablation); pause removal is what works.

8. Caveats

- One task, one policy family (ACT), 50 demos, sim.
- Fixed action-repeat approximates rate-matching; true per-frame timing is not recoverable from ISR output.
- 3 seeds: the 20-pt ISR–uniform gap is $\sim 2.5\times$ its standard error — solid direction, ± 8 pt magnitude.

9. Artifacts

Code, tables, per-cell eval_info.json, checkpoints and datasets:
Kavin60606/isr-aloha-transfer-cube-experiment (folder exp2/) and
Kavin60606/isr-aloha-act-ckpts (exp2/).