FAILURE-AWARE OFFLINE DISTILLATION FOR EFFICIENT VISION-LANGUAGE-ACTION MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

We present an offline distillation pipeline that transfers a large vision-language-action (VLA) teacher into a compact student, with one recipe across teachers, student backbones, and task suites: $\pi_{0.5}$ (3.3B) and UniVLA (7B) teachers, Long-CLIP (159M) and SigLIP2 (1.15B) students, and the four LIBERO suites. The pipeline has three parts. First, the student trains on success trajectories that the teacher collects once. Second, rule code labels teacher frames with eight behaviour phases from gripper commands and physical events recovered by replay, and the same rule code together with a vision-language model (VLM) judge finds the phase in which each student failure occurs, with a third model resolving their disagreements (rule code + VLM judge). These phases match an author's blind check on 46 of 47 failures. Third, the student is retrained on the same teacher data with twice the loss on the phases that hold at least one third of the failures on its suite. Phase labelling and phase weighting bring the students close to their teachers. Success rises on all 14 weighted configurations (125 episodes rescued, 68 broken, $p < 0.001$), and on LIBERO-10 the weighted Long-CLIP and SigLIP2 students reach 88.0% and 90.5%, against their UniVLA teacher's 91.5%. Training a weighted student, including its unweighted twin and the judge calls, costs 8.2 or 12.3 USD. At inference the students take 37 and 172 ms per forward pass on a 12 GB GPU that cannot hold UniVLA, against 431 ms for $\pi_{0.5}$.

1 INTRODUCTION

A robot runs its policy on a fixed compute budget, yet current generalist multi-task manipulation policies are VLA models with billions of parameters (Kim et al., 2024; Black et al., 2025a; Bu et al., 2025). On LIBERO-10 (Liu et al., 2023), the $\pi_{0.5}$ and UniVLA teachers succeed on 91% and 91.5% of 200 evaluation episodes. A compact student of 0.16 to 1.15B parameters, trained on data the teacher collects once, runs on a 12 GB GPU that cannot hold the 7B teacher. This paper studies two questions: how to distil a large VLA teacher into a small student from a fixed dataset, and where training effort should go when the student trails the teacher.

Existing work leaves three gaps. First, teacher-student methods for VLAs transfer through the teacher's architecture, its features, or its supervision of the student's online rollouts, so each procedure is tied to one teacher-student pair. RLDG (Xu et al., 2025) also transfers through a static dataset, but it distils reinforcement-learning specialists into a generalist rather than a large VLA into a compact student. Second, imitation learning chooses which frames matter from keyframes, gripper changepoints, VLM stage text, or learned data values. None of these looks at where the trained student fails, and the methods based on the gripper command treat a close command as a completed grasp even when the object is not held. Third, methods that feed failures back into policy training do so through a reward for a residual policy or through preference data, and both need new interaction or a new policy component.

We connect teacher and student through one static dataset, so the teacher appears once, during collection. The labelling and the diagnosis run in the simulator on the teacher trajectories and on recorded student rollouts, and they output loss weights on the same dataset; the student's observations and action head never change. Each teacher collects its own success trajectories into one fixed dataset D per suite (Figure 1), and every student trains on D with one training framework and one fixed schedule per backbone. Rules label the teacher frames with eight behaviour phases from the action

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

commands, the proprioceptive state, and the object states recovered by replaying the trajectory in the simulator. The same rules segment each failed student rollout recorded with the full simulator state, and a VLM judge attributes each failure independently. A third model resolves their disagreements from the recorded state and frames. Frames of the selected phases in D then receive twice the loss, and the student retrains from the original pretrained backbone. We compare the weighted and unweighted students on the same evaluation episodes.

Our contribution is one pipeline in three parts, in which the teacher appears in one static dataset and the students' failures decide where the training weight goes. The pipeline is cheap and comes close to the teacher: a student trains in 10 to 15 hours on one rented GPU for 4.1 to 6.2 USD, the judge calls of the diagnosis cost about 0.4 USD for all 14 weighted configurations, and with phase labelling and phase weighting the SigLIP2 students come within 1 point of their teachers on LIBERO-10.

- **Offline distillation across teachers, backbones, and suites.** One recipe trains a student in all 16 combinations from about 80K teacher frames per suite; after phase weighting, the 1.15B SigLIP2 students reach 90.0 and 90.5% on LIBERO-10, within 1 point of their 3.3B and 7B teachers.
- **Phase labelling and VLM-aided failure diagnosis.** Rule code labels teacher frames and student failures with eight phases defined by the gripper command and physical events recovered by replay; combining the rule code with a VLM judge (rule code + VLM judge) gives a phase that matches an author's blind check on 46 of the 47 checked failures, against 37 for the judge and 23 for the rules.
- **Weighting the phases where the student fails.** The phases that hold at least one third of the attributed failures receive twice the loss when the student retrains on the same teacher data; against the unweighted student, success rises on all 14 configurations, by 1 to 14 episodes.

2 Related Work

Compact policies and distillation. Some work makes an existing VLA cheaper to run: RLRC prunes and quantises it with a recovery stage (Chen et al., 2026b), and DeeR-VLA and MoLe-VLA reduce computation at inference by exiting the language model early (Yue et al., 2024) or by skipping layers (Zhang et al., 2026). SmolVLA (Shukor et al., 2025) and VLA-Adapter (Wang et al., 2026) instead design a small policy from the start and train it on human or community demonstrations; Table 1 lists the reported LIBERO results of SmolVLA, Octo (Ghosh et al., 2024), and VLA-OS (Gao et al., 2025). Distillation for VLAs transfers through different interfaces. RLDG fine-tunes a generalist on rollouts of reinforcement-learning specialists (Xu et al., 2025). ActDistill aligns a compact student to the teacher's layer-wise representations (Ye et al., 2025a), and VITA-VLA aligns a 7B VLM to the hidden states of a smaller pretrained action model (Dong et al., 2025); VLA-OPD queries teacher logits on the student's own trajectories (Zhong et al., 2026), and RPD supplies teacher actions during reinforcement learning (Jülg et al., 2025). Our students train only on recorded rollouts of a large generalist teacher, with no teacher features or online teacher queries, and one teacher dataset serves separately initialised compact backbones. Table 4 (Appendix A) compares the operating conditions of the closest methods.

Phases and stages in manipulation. Several methods split manipulation into segments, and most use them to structure demonstrations for the policy. AWE extracts waypoints where a trajectory segment is close to linear motion (Shi et al., 2023); UVD finds subgoals at phase shifts in a pretrained visual embedding (Zhang et al., 2024b); ManiLong-Shot splits a demonstration into interaction primitives at contact, grasp, and release events read from the robot state or proposed by a VLM (Chen et al., 2026c); and a phase-conditioned policy for deformable objects predicts its current phase at run time and starts a recovery trajectory when the predictor reads a contact failure (Chen et al., 2026a). Stage and keyframe work shapes training with such structure: Keyframe-Focused imitation learning weights the loss at expert action changepoints (Wen et al., 2021), StaKe derives stage and keyframe targets from gripper events (Xu et al., 2026), CASD distils stage annotations from an offline VLM into a frozen generator whose predictions condition the policy, with no VLM call at test time (Ding et al., 2026), and an empirical study compares stage interfaces for VLA fine-tuning (Ji, 2026). None of these methods uses its segments to attribute a trained policy's failures

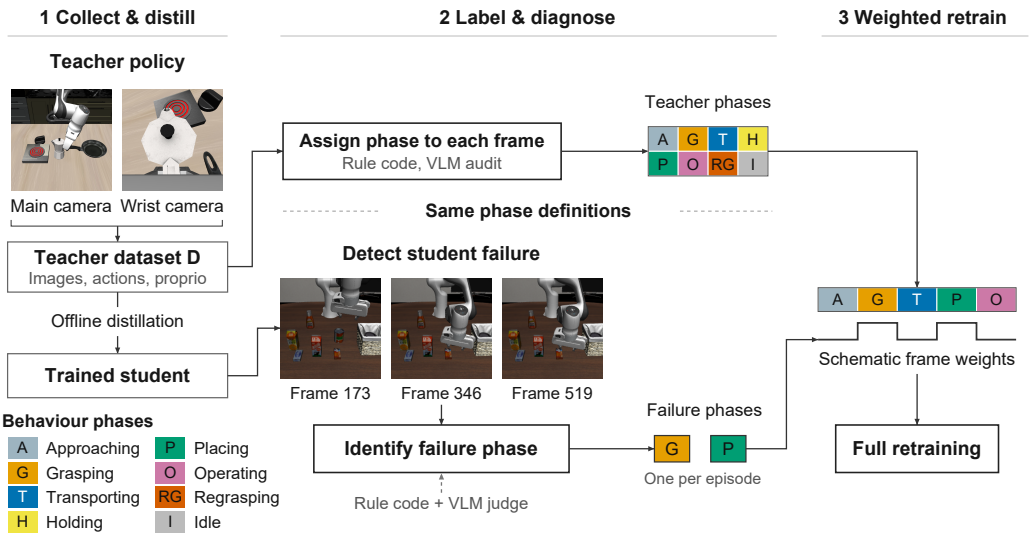

Figure 1: Pipeline overview. First, the teacher collects one static dataset D and the student trains on it offline. Second, rule code assigns phases to the frames of D and a VLM audits a sample. Rule code and a VLM judge apply the same phase definitions to find the failure phase of each failed student rollout. Third, the selected phases receive twice the loss and the student retrains on D from the pretrained backbone.

offline, and none lets those failures decide which segments training should stress. Balanced BC rebalances samples (Parekh et al., 2026) and DataMIL selects data by a learned estimate of its value to the downstream policy (Dass et al., 2026); our weights follow named phases instead. We define eight phases by the gripper command and physical events recovered by replay, such as the object moving with the hand, and record object loss and stalls in fields separate from the phase. One definition labels the teacher frames and the student's failed rollouts, and the attributed failures select the phases that receive weight.

Failure evidence as a training signal. AHA, RoboFAC, and Guardian detect and explain manipulation failures with a VLM (Duan et al., 2025; Ye et al., 2025b; Pacaud et al., 2025); RoboFAC also proposes corrections, and AHA's explanations refine rewards and plans in downstream systems. Two methods feed failures directly back into policy training. ReCoVLA has a VLM infer the failure mode and recovery stage and compiles them into a reward for a residual recovery policy trained with online reinforcement learning (Hu et al., 2026). FAR pairs the action chunks of a failed attempt, found by drops in a learned value, with critic-ranked samples of the current policy as preference data, and reuses successful retries for continual improvement (Hao et al., 2026). Our diagnosis attributes the student's failures to phases, and the selected phases receive twice the loss on the existing teacher dataset, with no new training data and no new policy component.

3 Method

Figure 1 shows the procedure end to end.

3.1 Offline distillation recipe

Teacher-collected data. Each teacher collects success trajectories on each suite. A frame holds one 256×256 image from the main camera and one from the wrist camera, one 7-dimensional action (end-effector translation increments, rotation increments, gripper command), the task instruction, and the proprioceptive state (end-effector pose and finger opening). The training target is the five-step action chunk that starts at the frame. The eight replay-labelled datasets (both teachers on every

suite) hold 75K to 81K frames each, 640,666 frames in total; the teacher rows of Table 1 give each teacher's own success on every suite.

Student and training framework. The student consists of a vision backbone, an action head, and an instruction encoder. The backbone is Long-CLIP (Zhang et al., 2024a) or SigLIP2 (Tschannen et al., 2025), fine-tuned with LoRA (Hu et al., 2022) (rank 8, alpha 16; Appendix G); the full students have 159M and 1.15B parameters, of which 9.1M and 14.2M train. The input is the last three main-camera frames, the current wrist image, and the instruction through the backbone's text tower. The head emits the five-step chunk with 256 bins per dimension; the loss is cross-entropy with the three rotation dimensions weighted by two. AdamW, learning rate 5×10^{-4} after 800 linear warm-up steps, batch 16. Long-CLIP trains 30 epochs and SigLIP2 10 epochs, and every run starts from the original pretrained backbone. Every experiment changes only the data, the backbone with its schedule, and, in Section 3.4, the frame weights (Table 8 in Appendix F).

3.2 Phase definition and offline labelling

Eight behaviour phases and a completion status. *Approaching*: the empty hand moves toward, or repositions relative to, an object or a mechanism, including detours and the retreat after a release. *Grasping*: from the close command until the object moves with the hand; failed attempts count. *Transporting*: carrying a grasped object. *Holding*: the hand holds the object and both stay still, away from the target. *Placing*: aligning, lowering, inserting, or releasing at the destination; a closed-gripper set-down counts. *Operating*: pulling a drawer, turning a knob, closing a door, pushing a plate. *Regrasping*: a second or later close attempt on an object the hand has already acquired. *Idle*: the still frames before the first motion and after completion. A phase describes what the arm is doing at that moment. Whether the object is in hand, lost, or misplaced, whether a placement has stalled, and whether the task is complete are separate fields that never override the phase; completion is a terminal status outside the phase vocabulary.

Labelling. Labelling runs once, offline, on the teacher data, in four steps.

Replay. Each teacher trajectory restarts from its recorded initial simulator state, and the simulator replays the recorded actions step by step. The replayed proprioceptive state must match the recording at every step (end-effector position within 10^{-4} m, orientation within 10^{-3} in quaternion distance, finger opening within 10^{-4} m) and stop at the same step; all trajectories of the eight labelled datasets pass. Replay recovers three quantities absent from the recording: the pose of every object, the contacts between the gripper, the objects, and the supports, and the target regions and mechanism joint angles of the task definition. Section 4.3 reports how often a VLM reviewer accepts the labels built on the recovered states.

Per-frame evidence. For every frame the rules compute the gripper command and the finger response; end-effector translation and rotation over a 9-frame window; relative and joint motion between hand and object; the lift of the object above its initial support and its position and descent relative to the target region; contact onset and loss; and mechanism joint change.

Phase rules. The phase rules are deterministic code: each behaviour proposes frames and a merge step resolves them. Grasping starts at the close command and lasts until the object and hand move together. Transporting requires an acquisition history and current in-hand evidence (bilateral contact, lift-off, joint motion of hand and object). Holding requires the object in hand, with hand and object still over the window and outside the target region. Placing starts when a held object enters the target region and descends, or enters the tight target region from any direction, and lasts until an observed release or the end of the recording; a placement that ends with the gripper closed counts as placing. Operating requires contact with a moving part and motion of a mechanism joint or a pushed object. Regrasping is a repeated close on an object with an acquisition history. Idle appears only in the still frames at the start and after completion. Short pauses inside a behaviour inherit it; the rules flag a placement that stays still for more than 20 frames as a stall. A final pass merges a segment of at most two frames into its neighbours when both neighbours carry the same phase. Appendix B gives the full rule set with its thresholds.

Output. Each labelled frame carries one behaviour phase plus in-hand, object-identity, loss, completion, and stall fields. The eight datasets hold 640,666 frames; Table 9 gives the phase composition

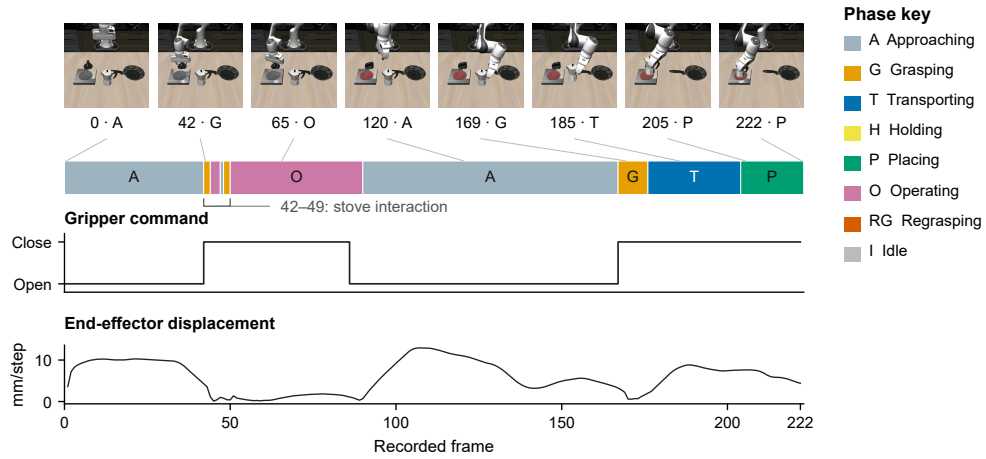

Figure 2: Phase timeline of one complete 223-frame UniVLA teacher trajectory on LIBERO-10. The teacher operates the stove, then places the moka pot. Below the strip, the gripper command and the per-step end-effector displacement are two of the rule inputs. The close command starts grasping, and descent into the target region starts placing.

of the LIBERO-10 and Goal datasets. Figure 2 shows one trajectory with its keyframes, its phase strip, and the rule inputs.

3.3 Student failure diagnosis

The diagnosis has three steps: the rules of Section 3.2 propose a failure phase from the recorded state of each failed rollout, a VLM judge attributes the same failure from the frames, and a third model resolves their disagreements from the recordings. A failure is attributed to the phase whose outcome the episode never reaches: a hand that stops at an object without closing, or an object that is never lifted, fails in grasping; an object lost in transit fails in transporting; an object released off the target fails in placing.

While the student acts on the evaluation episodes, the evaluator records every action and the full simulator state. For each failed rollout the rules of Section 3.2 segment the trajectory and propose the first anomaly: a stall of 20 steps in which the arm, the fingers, and every object stay still, a loss of the target object after the hand and object had moved together, or the step budget exhausted inside one phase. The phase that holds the first anomaly without a later recovery is the rule's attribution. An episode whose end-effector moves less than 1 cm and rotates less than 0.02 rad over the first 60% of the budget counts as a timeout without a phase.

VLM judge. The same failed rollouts go to a VLM judge (GPT-5.6-luna) that never sees the rule output. Each sample holds ten main-camera keyframes sampled uniformly in time, the wrist frames at the same steps, the gripper command sign at each of these steps, and the instruction. The prompt defines the phases with their reading rules and requires a JSON record with the failure phase, the failure mode, the step interval in which the key error first shows, the state of the object at that error, whether the robot recovered, and a confidence. The judge may answer uncertain.

Resolving disagreements. Where the rule and the judge disagree, or where either abstains, a third model (Claude) receives both attributions together with the recorded state and frames of that episode and assigns the phase by the definitions above. A judge mode such as wrong object maps to the phase in which it occurred. The resulting phase, marked rule code + VLM judge in the figures and tables, is the attribution the selection uses, and Section 4.3 reports the disagreements.

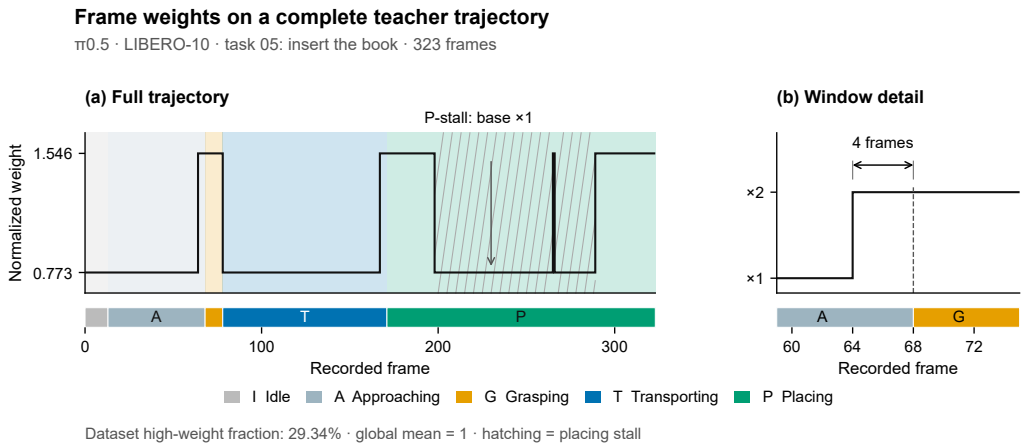

Figure 3: Frame weights over one teacher trajectory. The forward window raises the weight four frames before each selected phase. The placing stall returns to the base weight.

3.4 Weighted objective

Let L_t be the cross-entropy of the action chunk at frame t and w_t its weight; the objective is the batch mean of $w_t L_t$. The weights follow six steps in this order. (1) Regrasping merges into grasping. (2) For each task, a selected phase that covers more than 40% of the task's frames before dilation receives no weight on that task. (3) Every remaining frame of a selected phase receives the raw weight $r_t = 2$, all other frames $r_t = 1$, and placing frames flagged as a stall $r_t = 1$. (4) Inside each trajectory the weight dilates forward, $u_t = \max_{k=0,\dots,4} r_{t+k}$ over frames of the same episode, so that the frames whose action chunk starts before a selected phase also count. (5) Stall frames return to 1 after the dilation. (6) The weights divide by their mean over the dataset:

$$w_t = \frac{u_t}{\frac{1}{T}\sum_s u_s}. \tag{1}$$

After normalisation a selected frame carries twice the weight of a base frame, and the weighted and unweighted objectives share the same scale. Figure 3 shows the weight over one trajectory.

3.5 Which phases to weight

A phase receives weight on a suite when it holds at least one third of the attributed failures of the suite's unweighted students, with timeouts without a phase left out of the denominator. The rule selects grasping and placing on LIBERO-10, Goal, and Object, and grasping alone on Spatial. Table 6 (Appendix E) lists the shares.

4 Experiments

4.1 Setup

After the distillation results, we locate the phases where the students fail, weight them, measure the change in success, and check the diagnosis. We evaluate on the four LIBERO suites (Liu et al., 2023): 10 tasks per suite, 20 episodes per task, 200 episodes per configuration, and step limits of 520, 300, 280, and 220 for LIBERO-10, Goal, Object, and Spatial, as in the OpenVLA evaluation code (Kim et al., 2024), with last-epoch weights. Teachers and students run through the same evaluator and simulator version on the same 200 episodes of each suite, so the teacher's own success is the reference we compare against. We report the success rates of each weighted student and its unweighted twin; for every weighted configuration we also report the rescued episodes (unweighted fails, weighted succeeds), the broken episodes (the reverse), and an exact McNemar test (McNemar, 1947) on the discordant episodes. The attribution of Section 4.3 and the paired comparison of Section 4.4 share these 200 episodes. The selection passes the set of selected phases to training, and every weighted student retrains from the pretrained backbone on the teacher data.

Table 1: Mean success rate (%) on the four LIBERO suites. Teachers and students run 200 episodes per cell under one protocol, with last-epoch student weights. Weighted students retrain from the pretrained backbone with twice the loss on the phases of Section 3.5. They run on the same 200 episodes as their unweighted twin. The $\pi_{0.5}$ students on Spatial succeed on 198 and 199 episodes and have no weighted twin, shown as –. Bold marks the best student per column and teacher. The last three rows are compact policies trained or fine-tuned on LIBERO human demonstrations and evaluated under their own protocols. Their numbers come from OpenVLA (Kim et al., 2024) for Octo (Ghosh et al., 2024), VLA-Adapter (Wang et al., 2026) for VLA-OS (Gao et al., 2025), and the SmolVLA paper (Shukor et al., 2025).

Teacher → student	Spatial	Object	Goal	LIBERO-10
$\pi_{0.5}$ teacher	97.0	98.0	95.0	91.0
$\pi_{0.5}$ → Long-CLIP	99.0	96.5	91.0	81.5
$\pi_{0.5}$ → SigLIP2	**99.5**	97.0	94.0	86.5
$\pi_{0.5}$ → Long-CLIP, weighted	–	97.0	**98.0**	86.0
$\pi_{0.5}$ → SigLIP2, weighted	–	**98.5**	96.5	**90.0**
UniVLA teacher	95.0	95.0	94.0	91.5
UniVLA → Long-CLIP	91.5	96.0	95.5	87.0
UniVLA → SigLIP2	94.5	96.5	94.5	89.5
UniVLA → Long-CLIP, weighted	94.0	96.5	**96.5**	88.0
UniVLA → SigLIP2, weighted	**95.5**	**97.0**	96.0	**90.5**
Octo, 93M, human demonstrations (reported)	78.9	85.7	84.6	51.1
SmolVLA, 450M, human demonstrations (reported)	90	96	92	71
VLA-OS, 0.5B, human demonstrations (reported)	87.0	96.5	92.7	66.0

4.2 Distillation across teachers, backbones, and suites

Table 1 (plotted in Figure 6, Appendix F) gives the full matrix over two teachers, two backbones, and four suites. On Spatial and Object, seven of the eight unweighted student cells sit within 3 points of their teacher's success, and four of them exceed it, up to 99.0% (Long-CLIP) and 99.5% (SigLIP2) against 97.0% on Spatial. On Goal, every unweighted student sits within 4 points of its teacher, and the two UniVLA students exceed it. LIBERO-10 separates the students most, and the SigLIP2 student leads the Long-CLIP student for both teachers (86.5 against 81.5, 89.5 against 87.0). The weighted rows are the students of Section 4.4; after weighting the SigLIP2 students reach 90.0 and 90.5 on LIBERO-10, within 1 point of their teachers.

4.3 Where the student fails

Teacher-frame labels. The weights rest on the phase labels of the teacher frames. On 259 moments sampled from the LIBERO-10 and Goal datasets, a reviewer model (Claude Opus) accepts 94% of our replay-based labels, against 74% for labels from the gripper command alone, the signal that command-based stage and keyframe methods use (Table 5, Appendix C).

Failure phases of the unweighted students. Table 6 (Appendix E) attributes the failed episodes of the 14 unweighted students with a weighted twin to phases. Grasping or placing holds the largest share in every configuration, and the two together hold 67 to 100% of the attributed failures. The rules and the judge disagree on 32 of the 47 checked episodes (Table 7), and the most common disagreements follow three patterns, which the third model resolves from the recorded state and frames. The rule reports a lost object when the object slips from the gripper during lift-off or the gripper holds the wrong object (a failed or wrong grasp, attributed to grasping). The rule reports an approaching stall when the hand stops at the object without a close command (attributed to grasping, whose outcome the episode never reaches). The judge reads a cup set on the rim of a plate as placed while the simulator's target predicate does not (attributed to placing).

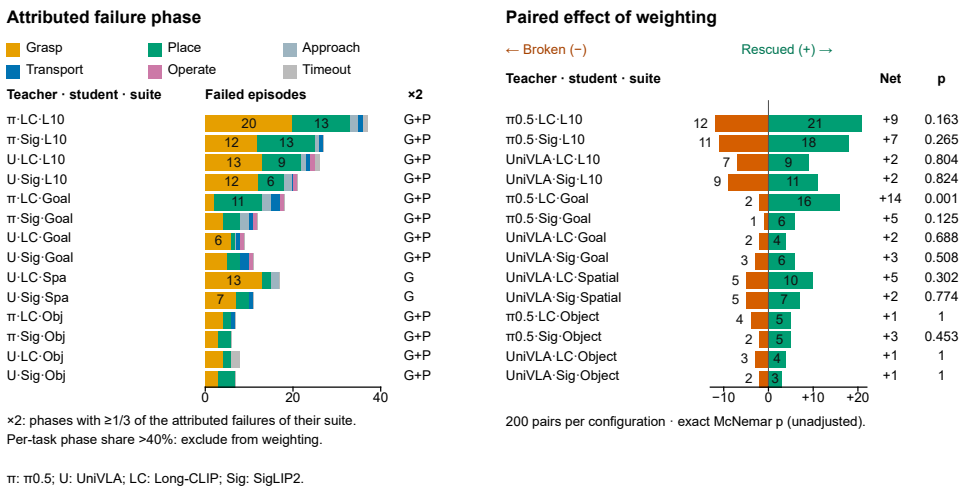

Figure 4: Failure phases and paired outcomes. The left panel shows attributed failure phases of the 14 students of Table 6 and the selected phases. Timeouts stay out of the selection denominator. The right panel shows rescued and broken episodes, net change, and exact McNemar p.

Table 2: Successes in 200 episodes for diagnosed and control phases. Letters abbreviate grasping (G), placing (P), transporting (T), and operating (O).

Configuration	Unweighted	Diagnosed phases	Control
$\pi_{0.5}$ → Long-CLIP, LIBERO-10	163	**172** (G, P)	166 (T, O)
$\pi_{0.5}$ → SigLIP2, LIBERO-10	173	**180** (G, P)	176 (T, O)
UniVLA → Long-CLIP, Spatial	183	**188** (G)	184 (G, P)
UniVLA → SigLIP2, Spatial	189	**191** (G)	188 (G, P)

4.4 Weighting the phases where the student fails

The 14 weighted students of Table 1 retrain with their selected phases at twice the loss and meet their unweighted twins on the same 200 episodes. Two configurations illustrate the paired counts: on $\pi_{0.5}$ with Long-CLIP on Goal, weighting rescues 16 episodes and breaks 2 (182 to 196 successes, $p = 0.001$); on $\pi_{0.5}$ with SigLIP2 on Object, it rescues 5 and breaks 2 (194 to 197; Figure 4, right). The weighted student gains on all 14 configurations, by 1 to 14 episodes; pooled over the 14, weighting rescues 125 episodes and breaks 68 (exact binomial $p < 0.001$). The three largest gains (182 to 196, 163 to 172, 173 to 180) fall on $\pi_{0.5}$ students, two of them on LIBERO-10.

4.5 Ablation

Two controls replace the phases the diagnosis selects (Section 3.5; Table 2). On the two $\pi_{0.5}$ students on LIBERO-10, weighting two phases the diagnosis does not select, transporting and operating, with a comparable budget of high-weight frames (25% against 29%), reaches 166 and 176 successes, against 172 and 180 for the diagnosed phases. On the two UniVLA students on Spatial, weighting grasping and placing, the pair the other suites select, reaches 184 and 188, against 188 and 191 for grasping alone; for SigLIP2 it falls below the unweighted 189. In all four configurations the diagnosed phases give the highest success, by 3 to 6 episodes.

Accuracy of the diagnosis. The third model settles each disputed episode. Figure 5 shows three of the contested episodes. On 48 failed episodes sampled across ten of the configurations (4 to 6 each), an author, blind to the rule, judge, and final phases, labelled the failure phase from the recorded keyframes and settled 47. The final phase matches that check on 46 (98%), the judge's on 37 (79%), and the rules' on 23 (49%). On the 32 episodes where the three sources do not all agree, 31, 22, and

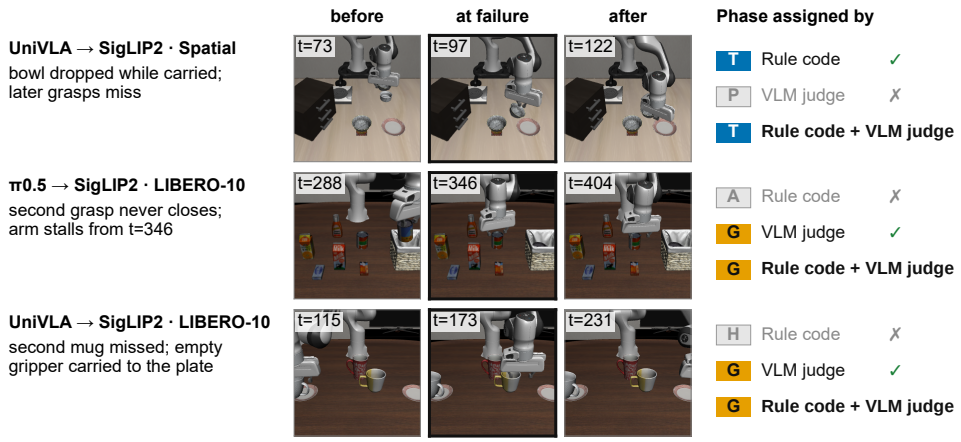

Figure 5: Three failed episodes on which the rule code and the VLM judge disagree. Each row shows the judge's main-camera keyframes before, at, and after the failure. The right column gives each source's phase. Rule code + VLM judge is the final phase, set by the third model from the recorded state and frames. The rule code is right on the first episode and the judge on the other two.

8 (Table 7). A second VLM (Claude), checking all 217 failed episodes, agrees with the final phase on 215 (99%).

4.6 Cost of training and inference

In Table 3, the Long-CLIP student trains on about 80K teacher frames in 9.9 hours on one rented RTX 4090 (48 GB), 4.1 USD at 0.41 USD per hour, and the SigLIP2 student in 15.0 hours, 6.2 USD; with the weighted retraining and the judge calls for all 217 failed episodes of Table 6 (about 0.4 USD at list prices), a weighted student costs about 8 or 12 USD in training compute and judge calls. On an RTX 3080 Ti (12 GB) the students peak at 1.6 and 5.1 GiB of GPU memory, against 9.7 GiB for the $\pi_{0.5}$ teacher, whose forward pass takes 431 ms against 37 ms for the Long-CLIP student and 172 ms for the SigLIP2 student; the UniVLA teacher cannot be deployed on this card at all, since its bf16 weights alone take 14 GiB.

Table 3: Training cost, model time, and memory. Student cost is the training time on an RTX 4090 (48 GB) at 0.41 USD per hour. The second number adds weighted retraining and its judge calls. Appendix H gives the sources of the teacher rows, and – marks values they do not report. Model time is the trimmed mean of 100 forward passes, with fp32 Long-CLIP, fp16 mixed-precision SigLIP2, and bf16 $\pi_{0.5}$. Peak memory covers six LIBERO-10 episodes. Both use one RTX 3080 Ti (12 GB).

Policy	Params	Training data	Training compute	Cost (USD)	Model time (ms)	Peak memory (GiB)
$\pi_{0.5}$ teacher	3.3B	10K+ h robot data; 273K LIBERO frames	–	–	431	9.7
UniVLA teacher	7B	OXE, GNM, Ego4D subsets; LIBERO demos	960 A100-h (pretraining)	–	–	14 (weights)
Long-CLIP student	159M	80K teacher frames	9.9 GPU-h (RTX 4090)	4.1 / 8.2	37	1.6
SigLIP2 student	1.15B	80K teacher frames	15.0 GPU-h (RTX 4090)	6.2 / 12.3	172	5.1
Diagnosis by the VLM judge, all 217 failed episodes of Table 6: about 0.4 USD at list prices						

5 Conclusion

This paper presents an offline distillation pipeline that turns a large VLA teacher into a compact student from one dataset the teacher collects once. The pipeline is flexible. It asks nothing of the teacher beyond successful rollouts, and one recipe holds across teachers, student backbones, and task suites. Phase labelling and phase weighting direct training to the phases where the student fails and bring it close to its teacher. A student costs 4.1 to 6.2 USD to train and needs a fraction of the teacher's memory and forward-pass time. This makes the pipeline a promising route to manipulation policies on edge devices, where compute and memory are scarce.

REFERENCES

Kevin Black, Noah Brown, James Darpinian, Karan Dhabalia, Danny Driess, Adnan Esmail, Michael Robert Equi, Chelsea Finn, Niccolo Fusai, Manuel Y. Galliker, Dibya Ghosh, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Devin LeBlanc, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Allen Z. Ren, Lucy Xiaoyang Shi, Laura Smith, Jost Tobias Springenberg, Kyle Stachowicz, James Tanner, Quan Vuong, Homer Walke, Anna Walling, Haohuan Wang, Lili Yu, and Ury Zhilinsky. $\pi_{0.5}$: a vision-language-action model with open-world generalization. In *Conference on Robot Learning (CoRL)*, 2025a.

Kevin Black, Noah Brown, Danny Driess, Adnan Esmail, Michael Equi, Chelsea Finn, Niccolo Fusai, Lachy Groom, Karol Hausman, Brian Ichter, Szymon Jakubczak, Tim Jones, Liyiming Ke, Sergey Levine, Adrian Li-Bell, Mohith Mothukuri, Suraj Nair, Karl Pertsch, Lucy Xiaoyang Shi, James Tanner, Quan Vuong, Anna Walling, Haohuan Wang, and Ury Zhilinsky. π_0: A vision-language-action flow model for general robot control. In *Robotics: Science and Systems (RSS)*, 2025b.

Qingwen Bu, Yanting Yang, Jisong Cai, Shenyuan Gao, Guanghui Ren, Maoqing Yao, Ping Luo, and Hongyang Li. UniVLA: Learning to act anywhere with task-centric latent actions. In *Robotics: Science and Systems (RSS)*, 2025. arXiv:2505.06111.

Dayuan Chen, Kai Tang, Yukuan Zhang, Kazuhiro Kosuge, and Yasuhisa Hirata. Phase-conditioned imitation learning with autonomous failure recovery for robust deformable object manipulation. *IEEE/ASME Transactions on Mechatronics*, 2026a. arXiv:2605.29407.

Yuxuan Chen, Yixin Han, Yize Huang, and Xiao Li. RLRC: Reinforcement learning-based recovery for compressed vision-language-action models. *IEEE Robotics and Automation Letters*, 11(7): 8864–8871, 2026b. arXiv:2506.17639.

Zixuan Chen, Chongkai Gao, Lin Shao, Jieqi Shi, Jing Huo, and Yang Gao. ManiLong-Shot: Interaction-aware one-shot imitation learning for long-horizon manipulation. In *AAAI Conference on Artificial Intelligence*, 2026c. arXiv:2512.16302.

Shivin Dass, Alaa Khaddaj, Logan Engstrom, Aleksander Madry, Andrew Ilyas, and Roberto Martín-Martín. DataMIL: Selecting data for robot imitation learning with datamodels. In *International Conference on Learning Representations (ICLR)*, 2026.

Tinghe Ding, Jiahao Li, and He Wang. CASD: Chunk-aligned semantic distillation for multi-stage robot manipulation. *arXiv preprint arXiv:2609.08638*, 2026.

Shaoqi Dong, Chaoyou Fu, Haihan Gao, Yi-Fan Zhang, Chi Yan, Chu Wu, Xiaoyu Liu, Yunhang Shen, Jing Huo, Deqiang Jiang, Haoyu Cao, Yang Gao, Xing Sun, Ran He, and Caifeng Shan. VITA-VLA: Efficiently teaching vision-language models to act via action expert distillation. *arXiv preprint arXiv:2510.09607*, 2025.

Jiafei Duan, Wilbert Pumacay, Nishanth Kumar, Yi Ru Wang, Shulin Tian, Wentao Yuan, Ranjay Krishna, Dieter Fox, Ajay Mandlekar, and Yijie Guo. AHA: A vision-language-model for detecting and reasoning over failures in robotic manipulation. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.00371.

Chongkai Gao, Zixuan Liu, Zhenghao Chi, Junshan Huang, Xin Fei, Yiwen Hou, Yuxuan Zhang, Yudi Lin, Zhirui Fang, and Lin Shao. VLA-OS: Structuring and dissecting planning representations and paradigms in vision-language-action models. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Dibya Ghosh, Homer Walke, Karl Pertsch, Kevin Black, Oier Mees, Sudeep Dasari, Joey Hejna, Tobias Kreiman, Charles Xu, Jianlan Luo, You Liang Tan, Lawrence Yunliang Chen, Quan Vuong, Ted Xiao, Pannag R. Sanketi, Dorsa Sadigh, Chelsea Finn, and Sergey Levine. Octo: An open-source generalist robot policy. In *Robotics: Science and Systems (RSS)*, 2024.

Haoran Hao, Shahram Najam Syed, Jeffrey Ichnowski, and Jeff Schneider. FAR: Failure-aware retry for test-time recovery and continual policy improvement. *arXiv preprint arXiv:2607.01111*, 2026.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations (ICLR)*, 2022. arXiv:2106.09685.

Haodi Hu, Chung-Ta Huang, Jing Liu, Ye Wang, Kei Suzuki, Matthew Brand, and Toshiaki Koike-Akino. ReCoVLA: VLM-Guided reward compilation for failure recovery in vision-language-action policies. *arXiv preprint arXiv:2606.09630*, 2026.

Yingwei Ji. An empirical study on stage-information interfaces for VLA fine-tuning. *arXiv preprint arXiv:2607.13605*, 2026.

Tobias Jülg, Wolfram Burgard, and Florian Walter. Refined policy distillation: From VLA generalists to RL experts. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2025. arXiv:2503.05833.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan Foster, Pannag Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. OpenVLA: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*, 2024. arXiv:2406.09246.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023. arXiv:2306.03310.

Quinn McNemar. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

Paul Pacaud, Ricardo Garcia, Shizhe Chen, and Cordelia Schmid. Guardian: Detecting robotic planning and execution errors with vision-language models. *arXiv preprint arXiv:2512.01946*, 2025.

Sagar Parekh, Heramb Nemlekar, and Dylan P. Losey. Towards balanced behavior cloning from imbalanced datasets. *Autonomous Robots*, 50(1):9, 2026.

Lucy Xiaoyang Shi, Archit Sharma, Tony Z. Zhao, and Chelsea Finn. Waypoint-based imitation learning for robotic manipulation. In *Conference on Robot Learning (CoRL)*, 2023.

Mustafa Shukor, Dana Aubakirova, Francesco Capuano, Pepijn Kooijmans, Steven Palma, Adil Zouitine, Michel Aractingi, Caroline Pascal, Martino Russi, Andres Marafioti, Simon Alibert, Matthieu Cord, Thomas Wolf, and Remi Cadene. SmolVLA: A vision-language-action model for affordable and efficient robotics. *arXiv preprint arXiv:2506.01844*, 2025.

Michael Tschannen, Alexey Gritsenko, Xiao Wang, Muhammad Ferjad Naeem, Ibrahim Alabdulmohsin, Nikhil Parthasarathy, Talfan Evans, Lucas Beyer, Ye Xia, Basil Mustafa, Olivier Hénaff, Jeremiah Harmsen, Andreas Steiner, and Xiaohua Zhai. SigLIP 2: Multilingual vision-language encoders with improved semantic understanding, localization, and dense features. *arXiv preprint arXiv:2502.14786*, 2025.

Yihao Wang, Pengxiang Ding, Lingxiao Li, Can Cui, Zirui Ge, Xinyang Tong, Wenxuan Song, Han Zhao, Wei Zhao, Pengxu Hou, Siteng Huang, Yifan Tang, Wenhui Wang, Ru Zhang, Jianyi Liu, and Donglin Wang. VLA-Adapter: An effective paradigm for tiny-scale vision-language-action model. In *AAAI Conference on Artificial Intelligence*, 2026.

Chuan Wen, Jierui Lin, Jianing Qian, Yang Gao, and Dinesh Jayaraman. Keyframe-focused visual imitation learning. In *International Conference on Machine Learning (ICML)*, 2021. arXiv:2106.06452.

Charles Xu, Qiyang Li, Jianlan Luo, and Sergey Levine. RLDG: Robotic generalist policy distillation via reinforcement learning. In *Robotics: Science and Systems (RSS)*, 2025. arXiv:2412.09858.

Yuan Xu, Yixiang Chen, Kai Wang, Jiabing Yang, Peiyan Li, Qisen Ma, Yan Huang, and Liang Wang. Improving vision-language-action model fine-tuning with structured stage and keyframe supervision. *arXiv preprint arXiv:2606.26801*, 2026.

Wencheng Ye, Tianshi Wang, Lei Zhu, Fengling Li, Guoli Yang, and Hengtao Shen. ActDistill: General action-guided self-derived distillation for efficient vision-language-action models. *arXiv preprint arXiv:2511.18082*, 2025a.

Zewei Ye, Weifeng Lu, Minghao Ye, Tao Lin, Shuo Yang, Junchi Yan, and Bo Zhao. RoboFAC: A comprehensive framework for robotic failure analysis and correction. *arXiv preprint arXiv:2505.12224*, 2025b.

Yang Yue, Yulin Wang, Bingyi Kang, Yizeng Han, Shenzhi Wang, Shiji Song, Jiashi Feng, and Gao Huang. DeeR-VLA: Dynamic inference of multimodal large language models for efficient robot execution. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2411.02359.

Beichen Zhang, Pan Zhang, Xiaoyi Dong, Yuhang Zang, and Jiaqi Wang. Long-CLIP: Unlocking the long-text capability of CLIP. In *European Conference on Computer Vision (ECCV)*, 2024a. arXiv:2403.15378.

Rongyu Zhang, Menghang Dong, Yuan Zhang, Liang Heng, Xiaowei Chi, Gaole Dai, Li Du, Dan Wang, Yuan Du, and Shanghang Zhang. MoLe-VLA: Dynamic layer-skipping vision language action model via mixture-of-layers for efficient robot manipulation. In *AAAI Conference on Artificial Intelligence*, 2026.

Zichen Zhang, Yunshuang Li, Osbert Bastani, Abhishek Gupta, Dinesh Jayaraman, Yecheng Jason Ma, and Luca Weihs. Universal visual decomposer: Long-horizon manipulation made easy. In *IEEE International Conference on Robotics and Automation (ICRA)*, 2024b.

Zhide Zhong, Haodong Yan, Junfeng Li, Junjie He, Tianran Zhang, and Haoang Li. VLA-OPD: Bridging offline SFT and online RL for vision-language-action models via on-policy distillation. *arXiv preprint arXiv:2603.26666*, 2026.

THE USE OF LARGE LANGUAGE MODELS

Vision-language models are components of the method, used for failure-phase attribution, label auditing and disagreement resolution as described in Sections 3.3 and 4.3, and no VLM output serves as an action target. Beyond these components, we used generative AI tools to draft and edit text, to write and edit LaTeX, and to write figure scripts. We have not used generative AI tools to generate synthetic data, to formulate or prove mathematical claims, to propose hypotheses, or to design the experiments. We have reviewed all AI-assisted work and take responsibility for the final content of this work, including text, claims and artifacts produced with the aid of generative AI.

A OPERATING CONDITIONS OF RELATED METHODS

B PHASE RULES AND THRESHOLDS

Replay acceptance. A replayed trajectory passes when, at every step, the end-effector position differs from the recording by less than 10^{-4} m, the orientation by less than 10^{-3} in quaternion distance, and the finger opening by less than 10^{-4} m, and when the replay terminates at the recorded step. All teacher trajectories and all recorded student rollouts used for rule diagnosis pass. The rules read the recorded proprioception, never the replayed one; replay supplies only object, contact, and target evidence.

Table 4: Training and deployment conditions of the closest methods, taken from the cited papers. "Teacher calls" counts teacher forward passes during student optimisation. "Environment" lists every stage that runs a simulator or robot. "Privileged access" lists inputs beyond images, actions, and proprioception.

Method	Student trains on	Teacher calls	Environment	Privileged access	Added at deployment
RLDG (Xu et al., 2025)	RL-specialist rollouts (actions)	none	specialist RL training and collection (real robot)	specialists: reward classifier, human interventions	none
ActDistill (Ye et al., 2025a)	teacher features on demos	per batch	none	teacher features	dynamic layer router
VLA-OPD (Zhong et al., 2026)	teacher logits on student rollouts	every step of every student rollout	student rollouts, optimisation	teacher logits	none
DataMIL (Dass et al., 2026)	target demos + prior data selected by datamodel scores	none	none (selection by held-out loss on target demos)	none	none
StaKe (Xu et al., 2026)	demos + stage/keyframe targets	none	none	none	none
CASD (Ding et al., 2026)	demos + VLM stage descriptions	none	none	offline VLM text	stage-context predictor
ReCoVLA (Hu et al., 2026)	residual RL on diagnosis-derived reward	none	online residual RL	VLM diagnosis, simulator state	online VLM failure detector, residual policies
Ours	teacher rollouts (actions), phase weights	none	teacher collection; replay for labelling; student rollouts for diagnosis	replayed and recorded simulator state; VLM diagnosis; a third model for rule-judge disagreements	none

Evidence parameters. End-effector translation and rotation use a centred 9-frame window; in-hand evidence uses the 4 preceding frames only. Thresholds: end-effector motion 1 mm per step and rotation 0.02 rad; object motion 0.5 mm; relative slip between hand and object 3 mm and relative rotation 0.03 rad; lift above the initial support 12 mm; descent 0.5 mm per step over at least 2 consecutive steps; object identity by the nearest object within 14 cm of the hand with a 2 cm margin to the runner-up; release when contact is absent for 2 steps and the separation exceeds 2 mm; holding after 8 quiet steps; a placement stall after 20 steps with end-effector motion below 0.5 mm and finger motion below 0.1 mm; mechanism motion above 10^{-5} m (prismatic) or 10^{-4} rad (hinge); the near-target region extends 2 cm beyond the target footprint, 20 cm above and 3 cm below it, and the tight region 2 cm beyond the target box in every direction; a mechanism counts as in reach within 20 cm of the hand. A segment of at most two frames merges into its neighbours only when both neighbours carry the same phase.

Conflict handling. When several behaviours propose the same frame, the merge step applies evidence guards. Ongoing grasping, placing, or operating evidence takes precedence over stillness, so a blocked grasp or placement stays in its phase instead of becoming idle or holding. Placement evidence (alignment, descent, insertion, release at the destination) refines transporting into placing; proximity to the near-target region alone does not. Regrasping refines grasping only when the same object has an acquisition history; a second close command alone does not. Holding requires the object in hand, measured low motion outside the target region, and the absence of any other ongoing interaction. Mechanism or pushed-object evidence takes precedence over the closed-gripper and speed heuristics. Completion is a status axis and never overrides a phase. A frame that no behaviour proposes stays unlabelled.

Mechanism tasks. Tasks with a mechanism (a drawer, a door, or a knob) use the joint angle of the mechanism from the task definition; contact with the handle before the joint moves counts as approaching, and the frames during which the joint moves count as operating.

Diagnosis events. The first-anomaly rule of Section 3.3 uses the same windows; its stall is the placement stall's 20-step window and thresholds applied in every phase, with every object still. For the loss event, the target object counts as acquired in a student rollout once a close command is

issued, the object lies within 15 cm of the hand, and hand and object move in the same direction by more than 1 cm with relative drift of at most 1 cm over at least three samples; a loss counts only after acquisition, and recovery counts only for the same object. An episode whose end-effector moves less than 1 cm, rotates less than 0.02 rad, and changes finger opening by less than 2 mm over the first 60% of the budget counts as a timeout without a phase.

C LABEL AUDIT

The audit draws a stratified sample per dataset over the four LIBERO-10 and Goal datasets: up to six moments from each phase, each the middle frame of a randomly drawn segment of that phase, up to two from the ends of episodes without a placing phase, and 31 further frames per dataset drawn from the frames that an earlier version of the rules left without a phase. Each moment shows the main and wrist frames at -8, -4, 0, $+4$, and $+8$ steps with its replay-based and gripper-command labels; the reviewer (Claude Opus) answers for each label whether it is acceptable under the definitions of Section 3.2 and states the reason. The reviewer accepts the replay-based label on 59 of 66 moments on $\pi_{0.5}$ LIBERO-10, 62 of 64 on UniVLA LIBERO-10, 59 of 64 on $\pi_{0.5}$ Goal, and 64 of 65 on UniVLA Goal (244 of 259, 94%). The 15 rejected moments lie at phase boundaries: still frames after completion labelled approaching instead of idle (4), a release inside an open drawer labelled operating instead of placing (3), a one-frame label inside a continuous phase (3), a drawer pull or knob turn labelled approaching or grasping instead of operating (3), and a phase running 8 to 12 frames past its boundary (2). On the same 259 moments the reviewer accepts 191 (74%) of the gripper-command labels; their errors are the count of close commands as completed picks and the use of command differences as motion. Table 5 gives the per-dataset counts.

Table 5: Model-assisted label audit. For each teacher dataset, the reviewer model judges the replay-based and the gripper-command label of every sampled moment. The label columns count accepted labels.

Dataset	Moments	Replay-based labels accepted	Gripper-command labels accepted
$\pi_{0.5}$ LIBERO-10	66	59 (89%)	46 (70%)
UniVLA LIBERO-10	64	62 (97%)	40 (63%)
$\pi_{0.5}$ Goal	64	59 (92%)	50 (78%)
UniVLA Goal	65	64 (98%)	55 (85%)
All	259	244 (94%)	191 (74%)

D DIAGNOSIS RECORDS

The per-episode records of the unweighted students of Table 6, with the rule attribution, the judge's record, the final phase, and the third model's evidence note, accompany the code.

E FAILURE ATTRIBUTION AND PAIRED OUTCOMES

Table 6 attributes the failed episodes of the 14 unweighted students of Section 4.3 to phases, and Figure 4 plots the attribution beside the paired outcomes of the 14 configurations.

Table 6: Failed episodes of the 14 unweighted students with a weighted twin, by attributed failure phase. Phases come from rule code + VLM judge, in which the third model settles rule-judge disagreements. Shares are taken over the attributed failures, leaving out timeouts without a phase. The last column gives the phases that the rule of Section 3.5 selects for each suite, with G for grasping and P for placing.

Teacher × backbone × suite	Failed	Grasp	Place	Approach	Transp.	Oper.	Timeout (no phase)	Grasp share	Place share	Selected
$\pi_{0.5}$ × Long-CLIP × LIBERO-10	37	20	13	2	1	0	1	56%	36%	G, P
$\pi_{0.5}$ × SigLIP2 × LIBERO-10	27	12	13	1	1	0	0	44%	48%	G, P
UniVLA × Long-CLIP × LIBERO-10	26	13	9	1	1	1	1	52%	36%	G, P
UniVLA × SigLIP2 × LIBERO-10	21	12	6	2	0	1	0	57%	29%	G, P
$\pi_{0.5}$ × Long-CLIP × Goal	18	2	11	2	2	1	0	11%	61%	G, P
$\pi_{0.5}$ × SigLIP2 × Goal	12	4	4	2	1	1	0	33%	33%	G, P
UniVLA × Long-CLIP × Goal	9	6	1	0	1	1	0	67%	11%	G, P
UniVLA × SigLIP2 × Goal	11	5	3	0	2	1	0	45%	27%	G, P
UniVLA × Long-CLIP × Spatial	17	13	2	2	0	0	0	76%	12%	G
UniVLA × SigLIP2 × Spatial	11	7	3	0	1	0	0	64%	27%	G
$\pi_{0.5}$ × Long-CLIP × Object	7	4	2	0	1	0	0	57%	29%	G, P
$\pi_{0.5}$ × SigLIP2 × Object	6	3	3	0	0	0	0	50%	50%	G, P
UniVLA × Long-CLIP × Object	8	4	2	0	0	0	2	67%	33%	G, P
UniVLA × SigLIP2 × Object	7	3	4	0	0	0	0	43%	57%	G, P

Table 7: Agreement of each diagnosis source with an author's blind check. The author checked 48 failed episodes sampled across ten of the configurations, 4 to 6 per configuration, and settled 47. The author labelled each episode from its recorded keyframes and the instruction. Disputed episodes are those on which the rule code, the VLM judge, and rule code + VLM judge do not all agree.

Source	All (47)	LIBERO-10 Long-CLIP (12)	LIBERO-10, Goal SigLIP2 (19)	Spatial, Object (16)	Disputed (32)
Rule code	23 (49%)	6	11	6	8
VLM judge	37 (79%)	11	11	15	22
Rule code + VLM judge (ours)	**46 (98%)**	**12**	**19**	15	**31**

F CONFIGURATION AND ADDITIONAL RESULTS

Table 8: Shared training framework and backbone-specific schedule.

Setting	Long-CLIP student	SigLIP2 student
Observation	3 main frames + wrist frame + instruction	same
Action target	5-step chunk, 7 dims, 256 bins per dim	same
Loss	cross-entropy, rotation dims $\times 2$	same
Adapter	LoRA rank 8, alpha 16	same
Optimiser	AdamW, lr 5×10^{-4}, batch 16	same
Schedule	800 linear warm-up steps, then cosine decay to 10^{-6}	same
Epochs	30	10
Parameters (full / trainable)	159M / 9.1M	1.15B / 14.2M

Table 9: Frames per phase in the four LIBERO-10 and Goal teacher datasets.

Dataset	Approach	Grasp	Transport	Place	Operate	Idle	Regrasp	Hold
$\pi_{0.5}$ LIBERO-10	37,800	5,783	18,651	14,477	2,542	1,550	334	155
UniVLA LIBERO-10	37,368	5,748	20,068	13,317	2,685	2,126	89	64
$\pi_{0.5}$ Goal	39,897	7,374	14,671	6,894	9,102	2,348	168	0
UniVLA Goal	40,203	6,554	13,506	6,093	8,271	5,775	60	0

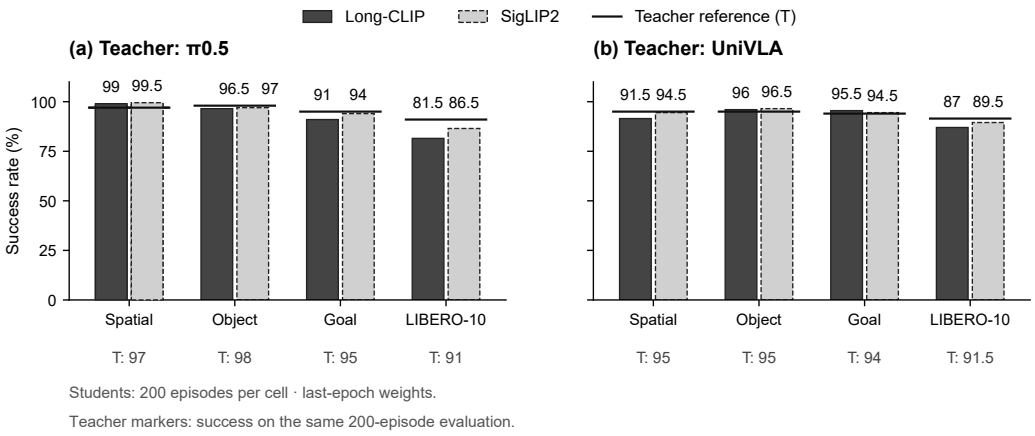

Figure 6: Success rate of every unweighted student from Table 1. Dark bars are Long-CLIP and light bars SigLIP2. Lines mark the teacher's own success on each suite under the same 200-episode protocol.

Weights. High-weight fraction after dilation: 29.3% ($\pi_{0.5}$ LIBERO-10), 28.1% (UniVLA LIBERO-10), 23.7% ($\pi_{0.5}$ Goal), 20.9% (UniVLA Goal); the normalised weights take the values 0.773 and 1.546 on $\pi_{0.5}$ LIBERO-10.

G STUDENT ARCHITECTURE AND TRAINING DETAILS

Both students share one design and one training program. Table 8 lists the shared settings and Table 10 the settings that differ between the two backbones.

Inputs and preprocessing. A training sample anchored at frame t of a teacher trajectory holds four RGB images, the main-camera frames $t-2$, $t-1$, and t followed by the wrist-camera frame t, together with the task instruction; its targets are the teacher actions $a_t, \ldots, a_{t+4}$ of the same trajectory. At test time the simulator renders both cameras at 256×256, and each image is rotated by $180°$, JPEG-encoded and decoded, and resized to 256×256 with a Lanczos filter, the same

Table 10: Backbone-specific settings of the two students. Table 8 gives the shared settings.

Setting	Long-CLIP student	SigLIP2 student
Pretrained checkpoint	Long-CLIP-B (ViT-B/16)	SigLIP2 So400m/16, 256 px
Image input	224×224	256×256
Embedding width d	512	1152
Instruction tokens	248	64
Head input ($6d$)	3072	6912
Trainable: head / adapters	8.27M / 0.86M	12.20M / 2.01M
Last-epoch learning rate	2.4×10^{-6}	1.3×10^{-5}
Test-time precision	float32	float16 mixed precision

steps that produced the stored teacher frames; at the start of an episode the first frame fills the history. Each backbone then applies its own image transform: Long-CLIP resizes to 224×224 and normalises with the CLIP channel mean and standard deviation, and SigLIP2 keeps 256×256 and maps pixel values to $[-1, 1]$. Training uses one augmentation, a brightness factor drawn uniformly from $[0.8, 1.2]$ for each of the four images independently. The proprioceptive state is not a student input.

Encoders and adapters. The four images pass separately through one shared vision tower, which returns one embedding per image, and the instruction passes through the text tower of the same model; every embedding is normalised to unit L2 norm. All pretrained weights of both towers stay frozen. The trainable parameters are the action head and the LoRA adapters of Table 8, which use dropout 0.1, leave the bias terms frozen, and start from LoRA's standard initialisation with a zero update. The Long-CLIP student starts from the Long-CLIP-B checkpoint, whose ViT-B/16 image tower and 12-layer text transformer share a 512-dimensional embedding space; its adapters sit on the attention output projection and the MLP output projection of every block of both towers; the Long-CLIP tokenizer maps the instruction to its 248-token context, and the text tower returns one 512-dimensional embedding. The SigLIP2 student starts from the So400m checkpoint with patch size 16 at 256×256, whose 27-layer vision tower returns the 1152-dimensional output of its pretrained attention-pooling head. The SigLIP2 (Gemma) tokenizer takes the lower-cased instruction, padded or truncated to 64 tokens, and the text tower returns one 1152-dimensional embedding. The SigLIP2 adapters sit on the query, key, value, and output projections of all 27 attention layers of the vision tower and on the output projection of the attention-pooling head; the SigLIP2 text tower carries no adapters.

Fusion and action head. With d the embedding width, the head input is the $6d$-dimensional concatenation $[e_{t-2};\, e_{t-1};\, e_t;\, e_t^{\text{wrist}};\, e^{\text{instr}};\, \mathbf{0}]$ of the three main-camera embeddings, the wrist embedding, the instruction embedding, and a constant zero vector of width d; fusion is this concatenation, with no attention across the inputs. The action head is a multilayer perceptron, $\text{Linear}(6d, 1024)$, LayerNorm, ReLU, dropout 0.1, $\text{Linear}(1024, 512)$, LayerNorm, ReLU, dropout 0.1, $\text{Linear}(512,\, 5 \cdot 7 \cdot 256)$, whose output reshapes into logits $z_{t,k,i} \in \mathbb{R}^{256}$ for chunk step $k \in \{0, \ldots, 4\}$ and action dimension $i \in \{1, \ldots, 7\}$.

Action discretisation and decoding. The actions keep the scale of the LIBERO controller commands, with no normalisation. Each of the seven dimensions (three translation increments, three rotation increments, the gripper command) is clipped to $[-1, 1]$ and discretised with the uniform binning of the OpenVLA action tokenizer (Kim et al., 2024), and the target $y_{t+k,i}$ is the bin of $a_{t+k,i}$. Decoding takes the arg-max class for every chunk step and dimension and returns the centre of its bin. The gripper dimension is discretised and decoded like the other six.

Chunking and test-time execution. The student runs at every policy step and predicts a chunk of $K = 5$ actions. With $\hat{a}_k^{(s)}$ the decoded entry k of the chunk predicted at step s, the executed action is the temporal ensemble of the chunks predicted at the current and the four preceding steps,

$$\bar{a}_t = \frac{\sum_{j=0}^{\min(t,4)} e^{-0.01j}\, \hat{a}_j^{(t-j)}}{\sum_{j=0}^{\min(t,4)} e^{-0.01j}}, \tag{2}$$

so the newest chunk receives the largest weight. The simulator executes $\bar{a}_t$, gripper value included, before the next policy step, and the instruction is tokenised once per episode. Each episode opens with ten steps of the action $(0, 0, 0, 0, 0, 0, -1)$ while the objects settle, as in the OpenVLA LIBERO evaluation, and the step limits of Section 4.1 count the policy steps that follow.

Loss and frame weights. For the sample at frame t,

$$L_t = \frac{1}{\sum_i \lambda_i} \sum_{i=1}^{7} \lambda_i \, \frac{1}{5} \sum_{k=0}^{4} \mathrm{CE}\big(z_{t,k,i},\, y_{t+k,i}\big), \qquad \lambda = (1, 1, 1, 2, 2, 2, 1), \tag{3}$$

where CE is the cross-entropy over the 256 classes and λ follows the order translation, rotation, gripper. A batch $\mathcal{B}$ gives the objective $\frac{1}{|\mathcal{B}|} \sum_{t \in \mathcal{B}} w_t L_t$ of Section 3.4, with w_t the normalised weight of the anchor frame t; the forward dilation of step (4) covers the same frames $t, \ldots, t + 4$ as the targets of the chunk. Unweighted students use $w_t = 1$, and the objective has no auxiliary term.

Optimisation. The optimiser sees 90% of the trajectories of each task, and the other 10% give a validation loss. AdamW with $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$, and weight decay 10^{-4} updates the adapters and the head at one shared learning rate. Over the first 800 optimiser steps the rate rises linearly from 5×10^{-6}, 1% of the peak, to 5×10^{-4}; from then on a cosine schedule, stepped once per epoch, gives epoch e of E the rate $\eta_e = 10^{-6} + \frac{1}{2}(5 \times 10^{-4} - 10^{-6})\big(1 + \cos(\pi(e-1)/E)\big)$, which ends at the values of Table 10. The gradient is clipped to a global L2 norm of 1.0 before every optimiser step, the samples are reshuffled every epoch, and training runs in 32-bit floating point. Every run starts from the original pretrained checkpoint with a new head and new adapters, and the evaluation uses the weights after the last epoch.

H TEACHER TRAINING SCALE

The teacher rows of Table 3 quote the original papers and repositories. UniVLA (Bu et al., 2025) pretrains its policy for 20,000 steps on 32 NVIDIA A100 GPUs in about 30 hours, which the paper reports as 960 A100-hours and as under one twentieth of the 21,500 A100-hours of OpenVLA (Kim et al., 2024); its data are subsets of Open X-Embodiment, GNM, and Ego4D, given as mixture weights, and the LIBERO checkpoints fine-tune for 30K to 40K steps at batch 128. The latent action model and the LIBERO fine-tuning add compute that the paper does not quantify. The $\pi_{0.5}$ base checkpoint of the official repository (`https://github.com/Physical-Intelligence/openpi`) is pretrained on more than 10,000 hours of robot data, with 280K pretraining and 80K post-training steps in the paper (Black et al., 2025a), and the LIBERO checkpoint fine-tunes for 30K steps at batch 256 on 273K frames; neither source reports the accelerators or the GPU-hours. $\pi_{0.5}$ keeps the transformer dimensions of π_0 (Black et al., 2025b), whose PaliGemma VLM and 300M action expert total 3.3B parameters.