Zero-Shot Utility and Efficient Adaptation in Vision-Language Multi-Agent Control

Daniel Masamba [1] **Christian I. Narcia-Macias** [1] **Erik Enriquez** [1] **Dongchul Kim** [1]

Abstract

Vision-language models (VLMs) can provide useful behavior before environment-specific training, but their role in partially observable multi-agent control remains unclear, especially when adapted from reinforcement-learning teachers. This work studies that question in PettingZoo Pursuit using controlled shifts in visual appearance, semantic mapping, observation layout, agent counts, and environment scale. The comparison includes zero-shot VLMs, LoRA-adapted VLM students distilled from a MAPPO teacher trained only on the base scenario, cold-start MARL, and trained IPPO/MAPPO baselines under matched local-observation constraints. The adapted VLM is therefore not treated as an independent alternative to MARL, but as a multimodal student whose performance reflects both teacher-policy distillation and pretrained visual-semantic priors. Results show that zero-shot VLMs provide non-trivial behavior relative to random cold-start RL, and that 1k–25k teacher demonstrations are sufficient for the adapted VLM to approach teacher-level base-task performance. Under visual and semantic shifts unseen by the teacher, the adapted VLM is more robust than trained MARL, while MAPPO remains strongest on the base task and some coordination-heavy variants. These results suggest a complementary division of strengths: MARL supplies task-specialized behavior, while VLM pretraining contributes visual-semantic invariances that can survive distillation.

[1]Department of Computer Science, The University of Texas Rio Grande Valley, Edinburg, TX, USA. Correspondence to: Daniel Masamba <daniel.masamba01@utrgv.edu>.

Proceedings of the 43^{rd} International Conference on Machine Learning, Seoul, South Korea. PMLR 306, 2026. Copyright 2026 by the author(s).

1. Introduction

Pretrained vision-language models (VLMs) offer a different starting point for multi-agent control: instead of learning perception and action entirely from random initialization, an agent can begin with multimodal representations acquired through large-scale pretraining. This is attractive when environment interaction is costly or task-specific data are limited. However, most evidence for VLM-based control still comes from single-agent settings or systems with substantial scaffolding. It remains unclear how useful these models are for decentralized, partially observable multi-agent tasks, where agents must act from local observations and coordinate without privileged global state.

This work studies that question in a controlled cooperative pursuit benchmark. The task is intentionally compact—an 8×8 PettingZoo Pursuit setting with local egocentric observations and a discrete 5-action space—but useful as a diagnostic testbed because it combines partial observability, local coordination, visual grounding, and controlled distribution shifts. The benchmark supports changes in visual appearance, semantic mapping, observation layout, agent counts, and environment scale, enabling comparison of base-task performance and robustness across controller families.

The comparison covers four regimes: zero-shot VLMs, LoRA-adapted VLMs distilled from a converged MAPPO teacher trained only on the base scenario, cold-start PPO-style MARL, and trained IPPO/MAPPO baselines. The adapted VLM should therefore not be interpreted as an independent alternative to MARL. It is a multimodal student of a task-specialized RL teacher, and its performance reflects both teacher-policy distillation and the pretrained backbone's visual-semantic priors. Similarly, offline demonstration budgets for VLM adaptation are not directly comparable to on-policy environment samples for MARL; the VLM scaling results measure student data efficiency given a fixed teacher cost.

This framing leads to three questions: (i) whether pretrained VLMs provide non-trivial behavior before task-specific training, using cold-start RL as a lower-bound reference; (ii) how efficiently a VLM student can absorb teacher behavior from offline demonstrations; and (iii) whether robustness un-

der visual and semantic shifts comes from the teacher policy or from the VLM backbone's pretraining priors. Because the teacher is trained only on the base scenario, robustness on unseen visual or semantic shifts cannot be attributed to teacher exposure to those shifts.

This paper makes three contributions:

- **A controlled student–teacher evaluation framework** for comparing zero-shot VLMs, distilled VLM students, cold-start MARL, and trained IPPO/MAPPO under matched local-observation constraints in partially observable cooperative pursuit.
- **Distillation and task-success scaling analyses** showing that zero-shot VLMs provide non-trivial behavior relative to random cold-start RL, and that LoRA-based distillation with 1k–25k teacher demonstrations allows the VLM student to approach teacher-level base-task performance, while validation accuracy and downstream task success can diverge under shift.
- **A robustness characterization across shift families** showing that trained MAPPO remains strongest on the base task and some coordination-heavy variants, whereas the distilled VLM is more robust to visual and semantic perturbations unseen by the teacher, suggesting that cross-shift transfer is primarily driven by pretrained visual-semantic priors.

2. Related Work

Embodied vision-language control. Recent work has shown that pretrained language and vision-language models can support embodied decision making when paired with appropriate grounding or action interfaces. SayCan combines language-model planning with affordance-weighted skill selection for robotic tasks (Ahn et al., 2022). PaLM-E conditions a multimodal language model on continuous sensor observations and studies transfer across embodied and non-embodied tasks (Driess et al., 2023). RT-2 adapts vision-language pretraining for action generation in robotic control (Zitkovich et al., 2023), while VIMA studies multimodal prompting and systematic generalization across task variations (Jiang et al., 2023). These works demonstrate the value of pretrained multimodal representations for embodied control, but most focus on single-agent settings or settings with substantial task scaffolding. In contrast, this paper studies a compact decentralized multi-agent setting where each agent acts from local observations. The goal is not to claim broad VLM superiority, but to diagnose what pretrained multimodal priors contribute before and after distillation from a MARL teacher.

Multi-agent reinforcement learning under partial observability. Cooperative MARL provides strong task-specialized methods for decentralized partially observable problems. The Dec-POMDP framework formalizes settings where agents must act from local observations while optimizing a shared objective (Oliehoek & Amato, 2016). PPO-based methods such as MAPPO have become strong baselines across cooperative benchmarks (Yu et al., 2022). MAPPO extends PPO to cooperative multi-agent settings by training decentralized policies with a centralized critic, allowing the value function to use joint or global information during training while each agent acts from its own local observation at execution time. This centralized-training/decentralized-execution structure makes MAPPO a natural teacher and task-specialized baseline for partially observable cooperative pursuit. Because MAPPO learns directly from task rollouts, it can acquire strong coordination behavior on the training distribution, but this specialization may also make it sensitive to visual or structural shifts not seen during training. These methods can perform very well after sufficient interaction, but training from scratch is often sample-intensive and can be sensitive to distribution shift (Amato, 2024). This motivates a complementary comparison with pretrained VLMs. In the present setting, however, the adapted VLM is distilled from a trained MAPPO teacher, so it should be interpreted as a multimodal student of MARL rather than an independent replacement for MARL. This student–teacher framing is central to the analysis.

Benchmarks, robustness, and multimodal-agent evaluation. A central challenge in evaluating multimodal agents is that success on a default task can hide brittle behavior under small visual, semantic, or structural changes. PettingZoo provides standardized multi-agent environments, including pursuit-style tasks that support controlled studies of coordination under partial observability (Terry et al., 2021). VIMA similarly emphasizes systematic task variation for embodied multimodal control (Jiang et al., 2023). PettingZoo Pursuit is used here as a diagnostic benchmark rather than as a claim of real-world visual complexity: the observations are synthetic and compact, but the environment allows controlled shifts in color, shape, layout, agent counts, and grid size. This makes it useful for separating three questions: whether VLMs provide non-trivial zero-shot behavior, how efficiently a VLM student absorbs teacher behavior, and whether robustness under visual-semantic shifts comes from the teacher policy or from the pretrained backbone.

3. Problem Setting

The setting is cooperative pursuit in a partially observable grid world. In the base scenario, two pursuers and one evader interact on an 8×8 PettingZoo Pursuit grid. Each pursuer receives only a local egocentric observation and selects one of five discrete actions: left, right, down, up, or

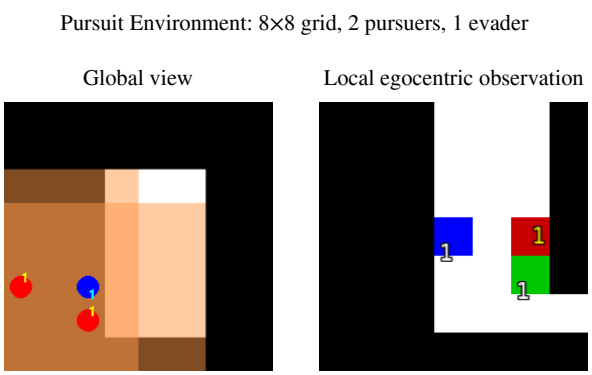


Figure 1. Pursuit environment. **Left:** global view of the base scenario rendered by PettingZoo, with two pursuers and one evader on an 8×8 grid; the lighter rectangle indicates one pursuer's local observation window. **Right:** the same agent's local egocentric observation, the only signal available to either VLM or RL controllers in the experiments. Blue marks the ego agent, green marks teammates, red marks evaders, and black marks walls or out-of-grid cells.

stay. Episodes are bounded by a fixed maximum horizon, and success is measured by whether the pursuers capture the evader within the episode. This setting is intentionally compact and synthetic, but it is useful for controlled analysis because agents must act from partial local information and coordinate without access to privileged global state (Figure 1).

The evaluation uses a shared local-observation interface for both VLM controllers and RL policies (Figure 2). Each controller acts from the same underlying local grid window rather than from a privileged global state. For VLM controllers, this local observation is rendered as an image and paired with a task prompt. For RL baselines, the same local information is provided through the policy interface. Thus, "matched" refers to information content rather than identical pixels: the VLM and obs-image RL baselines use different render scales and overlays (Sections A.2 and A.3), but all controllers receive the same partial state variables relevant to the task.

The evaluation covers four controller regimes: zero-shot VLMs, LoRA-adapted VLM students distilled from a converged MAPPO teacher, cold-start PPO-style MARL, and trained IPPO/MAPPO baselines. The MAPPO teacher used for distillation is trained only on the base scenario. Therefore, any robustness shown by the adapted VLM on unseen visual or semantic shifts cannot be attributed to teacher exposure to those shifts; it must come from the VLM backbone's pretrained visual-semantic priors, from coverage in the base-scenario demonstrations, or from the interaction between the two.

The benchmark supports controlled robustness tests through changes in visual appearance and task structure. These shifts are grouped into three families used throughout the analysis: *training-like* shifts, including the base task, moving evaders, and additional pursuers; *visual/semantic* shifts, including color shift, semantic color swap, and shape shift; and *structural* shifts, including larger grids, additional evaders, and non-centered egocentric observations. This organization supports comparison of controller families not only by base-task success, but also by how their behavior changes when perception, semantics, or task structure is perturbed.

4. Methods

4.1. Controller Families

Zero-shot VLMs. Instruction-prompted VLM controllers are evaluated without environment-specific fine-tuning. Each controller receives a rendered local observation and a task prompt, then outputs one of five discrete actions. Zero-shot results are reported for LLaVA-v1.6 (Liu et al., 2023) and Qwen-VL (Bai et al., 2023). LLaVA-v1.6 is the backbone used for all adapted-VLM experiments because the distillation pipeline—dataset format, action-token interface, AWR objective, and evaluation harness—was developed for this model. Qwen-VL is included only as a stronger zero-shot reference point, allowing the zero-shot signal to be checked against a stronger pretrained VLM rather than only LLaVA-v1.6. Qwen-VL is not adapted in this study because doing so would require a separate processor, chat template, and action-token implementation, while the central question is how one committed VLM backbone behaves as a student of a MARL teacher.

Adapted VLMs as MAPPO students. The adapted controller is a LLaVA-v1.6 model distilled from a converged MAPPO teacher using LoRA (Hu et al., 2022). The teacher is trained only on the base scenario, without domain randomization or exposure to the visual, semantic, or structural shifts used at evaluation time. The trained MAPPO baseline in Table 2 is a separately trained MAPPO policy of the same family, not the exact teacher checkpoint used for distillation; full training details are provided in Section A. The adapted VLM is therefore interpreted as a multimodal student of MARL rather than as an independent alternative to MARL. Its base-task performance reflects how well teacher behavior is absorbed, while its cross-shift behavior reflects the interaction between teacher demonstrations and the VLM backbone's pretrained visual-semantic priors.

Adapted VLMs are trained with 1k, 5k, 25k, and 100k offline teacher demonstrations. Each demonstration contains the local observation, teacher action, and value-related annotations. The main adapted results use advantage-weighted regression (AWR (Peng et al., 2019)), which gives higher weight to teacher actions with larger estimated advantage. This setup measures how offline teacher supervision scales

and whether stronger imitation of the base teacher improves or harms robustness under shift.

RL baselines. The comparison includes PPO-style MARL baselines trained with IPPO (de Witt et al., 2020) and MAPPO (Yu et al., 2022). Cold-start variants use random initialization with no training and serve only as lower-bound references for zero-shot behavior. Trained variants use up to 10M on-policy environment samples and serve as task-specialized baselines. For scaling analyses, MAPPO is also evaluated at intermediate training checkpoints.

4.2. Training Flow

The adapted VLM is trained offline from a fixed dataset of value-annotated demonstrations collected from the MAPPO teacher in the base environment. For each example, the teacher provides an action and value-related quantities such as rewards, returns, value estimates, and advantages. The local observation is rendered as an image and paired with the task prompt. No environment interaction occurs during VLM adaptation, so the training distribution is fixed by the teacher's base-scenario behavior.

The adapted VLM uses the AWR-style objective

$$\mathcal{L}_{\mathrm{AWR}}(\theta) = -\mathbb{E}_{\mathcal{D}}\left[w(A)\log \pi_\theta(a \mid x(o), p)\right],$$
$$w(A) = \mathrm{clip}\left(e^{A/\beta}, w_{\max}\right).$$

Here, $\mathcal{D}$ contains tuples (o, a, A), where o is the local observation, $x(o)$ is its rendered image, p is the task prompt, a is the MAPPO teacher action, and A is the teacher-derived advantage. The weight $w(A)$ increases the influence of teacher actions estimated to be more valuable.

The MARL baselines are trained on-policy. At each iteration, the policy collects new rollouts, estimates advantages from the resulting trajectories, and updates the actor–critic policy using PPO. Thus, the VLM consumes 1k–100k fixed offline teacher demonstrations, whereas MAPPO/IPPO consume up to 10^7 on-policy environment samples. These are not unit-equivalent measures of compute or supervision; the curves are interpreted as method-specific scaling profiles rather than direct sample-efficiency comparisons. Hyperparameters, rendering details, and compute settings are reported in Section A.

4.3. Action Decoding

Because VLM controllers emit text, the mapping from generated text to environment action is part of the method. Deterministic decoding is used throughout, with the same protocol applied across seeds, scenarios, and demonstration budgets.

Action vocabulary and prompt. The environment exposes five discrete actions: 0=left, 1=right, 2=down, 3=up, and 4=stay. Each VLM receives a rendered local observation and a prompt that specifies the agent role, goal, visual legend, action vocabulary, and output format. The base-scenario prompt is kept fixed:

```
You are pursuer `{agent_name}`.
Your goal is to catch the
red evader.  Blue=you,
Green=allies, Red=evader,
Black=walls.  Valid actions:
0=left,1=right,2=down,3=up,4=stay.
Choose the best action (0--4).
```

For shifted scenarios, only the legend or minimal task hint is changed when needed, e.g., to describe a new color palette, shape encoding, moving evaders, or a different number of pursuers. The action vocabulary and final instruction are held fixed. Chain-of-thought prompting and multi-turn dialogue are not used.

Adapted LLaVA decoding. For the LoRA-adapted LLaVA-v1.6 controller, constrained logit scoring is used rather than free generation. After applying the model's chat template, a deterministic forward pass scores only the tokens corresponding to valid action digits. For each action a, let $\mathcal{T}(a)$ be the token IDs corresponding to the bare and space-prefixed digit encodings of a. The selected action is

$$a^\star = \arg\max_{a \in \mathcal{V}(o)} \max_{i \in \mathcal{T}(a)} \ell_i,$$

where ℓ_i is the next-token logit and $\mathcal{V}(o)$ is the valid-action set for observation o. This valid-action set excludes moves into walls or out-of-grid cells; stay is always valid. Decoding is greedy, with no sampling, temperature, or top-p.

Zero-shot VLM decoding. For zero-shot LLaVA-v1.6 and Qwen-VL, the same prompt is used, but deterministic free generation is allowed. The output is parsed using a fixed sequence of rules: first, remove any `<think>...</think>` blocks; second, search for a labeled digit after words such as `action`, `move`, `answer`, or `choose`; third, search for English action words and synonyms such as left/right/up/down/stay or west/east/north/south; and finally, fall back to the first valid digit in the output. If no valid action is found, the action is recorded as invalid and replaced with stay. This protocol keeps zero-shot evaluation close to normal VLM prompting while making output parsing reproducible.

Evaluation protocol. All controllers are evaluated on the base scenario and the shift suite defined in Section 3. Performance is reported as success rate over 100 episodes and 3 random seeds. The same local-state information is used across VLM and RL controllers, so differences reflect controller family, training regime, and input encoding rather than privileged state access.

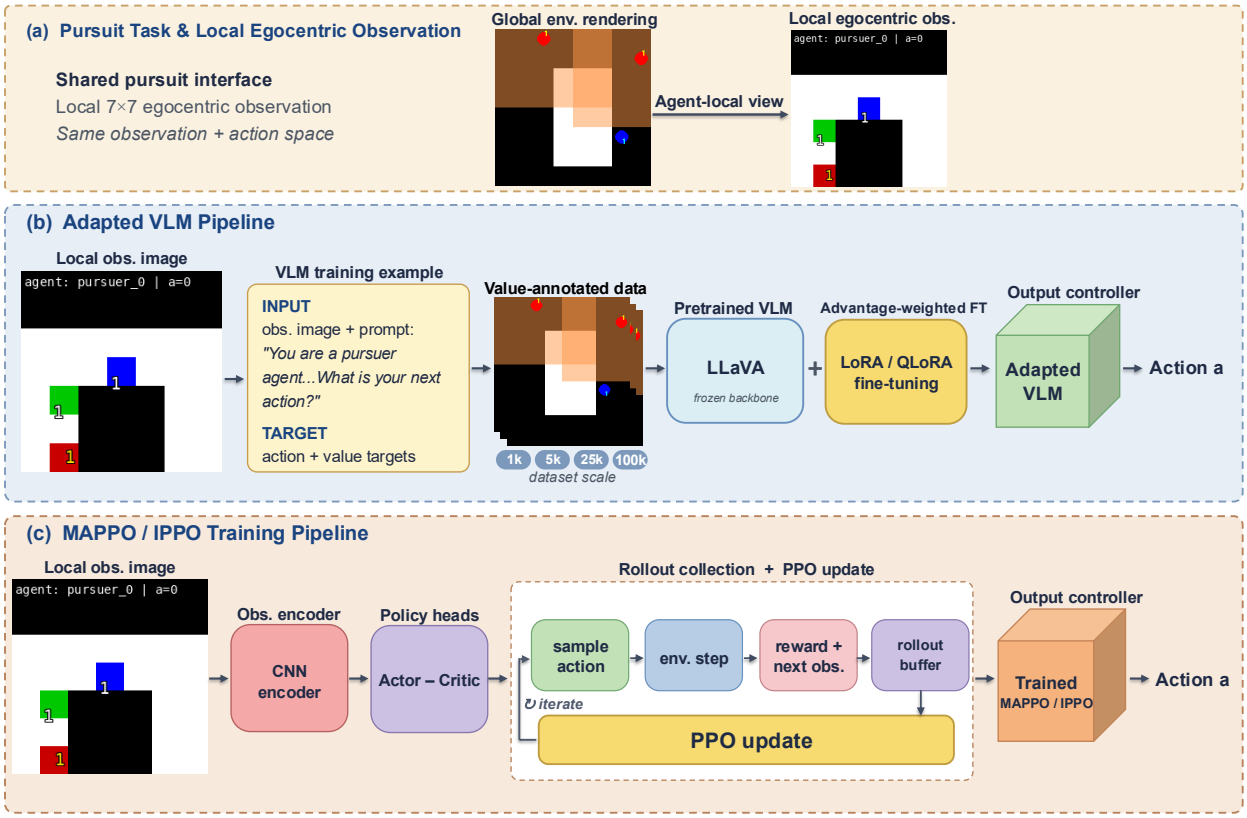


Figure 2. Training pipelines for VLM and MARL controllers in cooperative partially observable pursuit. **(a)** Both families act from matched local egocentric observations rather than privileged global state. **(b)** The VLM pipeline distills MAPPO-teacher actions and value/advantage annotations into offline vision-language datasets at 1k–100k scales, then adapts a pretrained LLaVA backbone with LoRA/QLoRA (Hu et al., 2022; Dettmers et al., 2023) using advantage-weighted fine-tuning. **(c)** The MARL pipeline trains IPPO/MAPPO actor–critic policies from scratch through on-policy rollout collection and PPO updates. The two pipelines consume different forms of supervision: fixed teacher demonstrations for VLM distillation and live environment interaction for MARL.

5. Results

5.1. Zero-Shot Utility Before Task-Specific Training

Table 1 compares instruction-prompted VLM controllers with cold-start PPO-style MARL policies. Cold-start MAPPO and IPPO are included as random-policy lower-bound references rather than as strong baselines. Zero-shot VLMs provide non-trivial behavior before task-specific training: Qwen-VL reaches 0.29 success on the base scenario and 0.74 with moving evaders, while cold-start RL remains at or below 0.05 across all scenarios. LLaVA-v1.6 is weaker than Qwen-VL in the zero-shot setting but still generally exceeds cold-start RL. These results indicate that pretrained multimodal representations provide some immediately usable signal in this diagnostic pursuit task, motivating their use as starting points for adaptation rather than as final controllers.

Table 1. Zero-shot VLMs versus cold-start RL. Entries are mean$_{\pm \text{std}}$ success rates over 100 episodes × 3 seeds. Cold-start RL is included as a random-policy lower-bound reference.

Scenario	LLaVA ZS	Qwen ZS	MAPPO CS	IPPO CS
Base	$0.12_{\pm 0.32}$	$\mathbf{0.29}_{\pm 0.45}$	$0.02_{\pm 0.12}$	$0.03_{\pm 0.16}$
Moving evaders	$0.64_{\pm 0.48}$	$\mathbf{0.74}_{\pm 0.44}$	$0.05_{\pm 0.21}$	$0.03_{\pm 0.17}$
Grid 16×16	$0.02_{\pm 0.13}$	$\mathbf{0.22}_{\pm 0.41}$	$0.01_{\pm 0.05}$	$0.01_{\pm 0.05}$
2 evaders	$0.04_{\pm 0.20}$	$\mathbf{0.08}_{\pm 0.27}$	$0.00_{\pm 0.03}$	$0.00_{\pm 0.00}$
3 pursuers	$0.05_{\pm 0.20}$	$\mathbf{0.21}_{\pm 0.41}$	$0.00_{\pm 0.00}$	$0.00_{\pm 0.00}$
Color shift	$0.13_{\pm 0.32}$	$\mathbf{0.28}_{\pm 0.45}$	$0.04_{\pm 0.20}$	$0.01_{\pm 0.06}$
Sem. color swap	$0.12_{\pm 0.31}$	$\mathbf{0.34}_{\pm 0.47}$	$0.04_{\pm 0.20}$	$0.01_{\pm 0.08}$
Shape shift	$0.11_{\pm 0.30}$	$\mathbf{0.34}_{\pm 0.47}$	$0.05_{\pm 0.21}$	$0.01_{\pm 0.09}$
Non-centered ego	$0.01_{\pm 0.06}$	$\mathbf{0.19}_{\pm 0.39}$	$0.03_{\pm 0.16}$	$0.03_{\pm 0.17}$

5.2. Adaptation Scaling from Offline Demonstrations

Figure 3 separates two quantities that are easy to conflate: validation accuracy and downstream task success. Validation accuracy is teacher-action prediction on a fixed held-out set of MAPPO demonstrations; task success is measured by running the controller in the environment. This distinction is

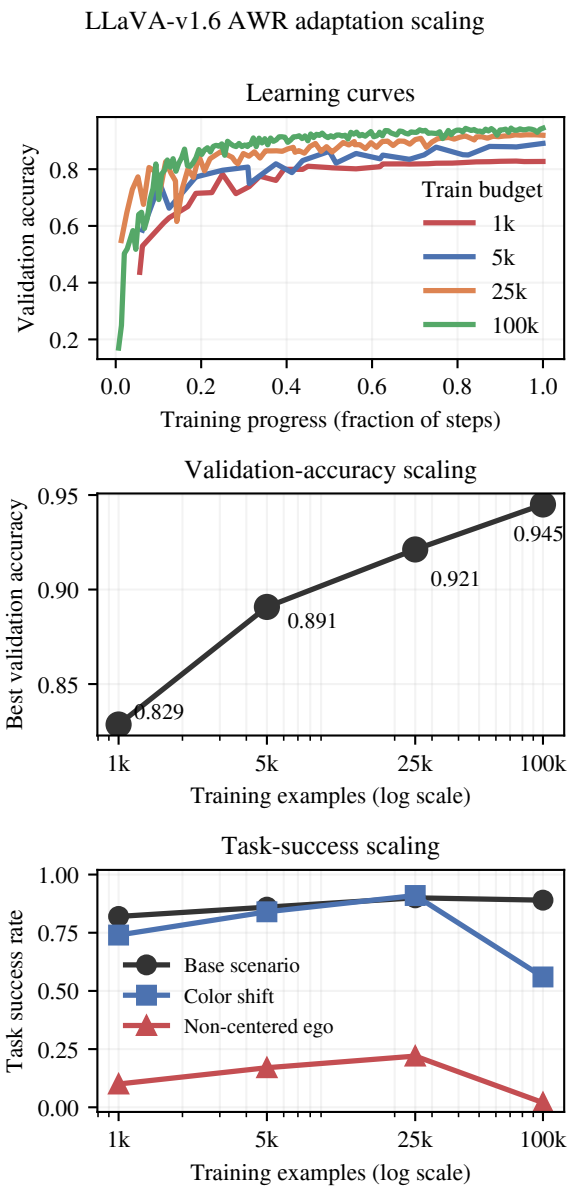


Figure 3. LLaVA-v1.6 AWR adaptation scaling. Top and middle: validation accuracy on held-out MAPPO-teacher demonstrations increases monotonically with budget. Bottom: downstream task success for representative scenarios from Table 2; transfer under shift is non-monotonic even when validation accuracy improves.

important because higher agreement with the base-scenario teacher does not necessarily imply better transfer under shift.

Validation accuracy increases monotonically from 0.829 with 1k demonstrations to 0.891 with 5k, 0.921 with 25k, and 0.945 with 100k. Thus, most of the held-out teacher-action prediction performance is recovered by 25k demonstrations, with diminishing returns beyond that scale.

Task success follows a different pattern. On the base scenario, success improves from 0.82 at 1k to 0.90 at 25k and remains similar at 100k. Under shift, however, larger in-distribution demonstration budgets can hurt transfer: color-shift success rises to 0.91 at 25k but drops to 0.56 at 100k, while non-centered ego rises to 0.22 at 25k and drops to 0.02 at 100k. This divergence suggests that stronger imitation of the base teacher can improve in-distribution agreement while eroding robustness-relevant visual or layout invariances.

5.3. Robustness After Adaptation or Training

Table 2 summarizes performance after VLM adaptation or RL training. On the base scenario, trained MAPPO and IPPO reach 1.00 success, while the best adapted VLM reaches 0.90. This is expected under the student–teacher framing: the adapted VLM is distilled from a MAPPO teacher and should not be interpreted as an independent learner surpassing MARL on the teacher's training distribution.

The pattern changes under visual and semantic perturbations. On color shift, the best adapted VLM reaches 0.91, compared with 0.63 for trained MAPPO and 0.00 for trained IPPO. On semantic color swap and shape shift, the best adapted VLM reaches 0.61 and 0.75, while trained MAPPO drops to 0.13 and 0.15. Because the distillation teacher is trained only on the base scenario, this robustness is better interpreted as evidence that pretrained VLM visual-semantic priors survive distillation, not as evidence that the teacher policy itself covers these shifts. The IPPO collapse under color and semantic shifts is consistent with strong specialization to the training appearance distribution and is best viewed as a failure mode of the trained baseline rather than as evidence that cold-start IPPO is preferable.

Structural shifts are more mixed. On the larger grid, trained MAPPO reaches 0.63 and outperforms the adapted VLM. On additional evaders, the best adapted VLM reaches 0.67, close to trained MAPPO at 0.64. Non-centered egocentric observation remains difficult for all methods, with the best adapted VLM at 0.22 and trained MAPPO at 0.01, indicating that observation-layout dependence remains a shared weakness.

Statistical reliability. With $N = 300$ trials per cell, the standard error of each mean is at most $0.50/\sqrt{300} \approx 0.029$, and the standard error of a difference between two budgets is at most $\sqrt{2} \cdot 0.029 \approx 0.041$. The color-shift drop from $0.91_{\pm 0.29}$ at 25k to $0.56_{\pm 0.50}$ at 100k ($\Delta = -0.35$) and the non-centered-ego drop from $0.22_{\pm 0.41}$ to $0.02_{\pm 0.13}$ ($\Delta = -0.20$) are therefore much larger than the sampling-noise scale. In contrast, the base-scenario difference between 25k and 100k is only 0.01 and should not be interpreted as meaningful. A plausible explanation is that larger

Table 2. Generalization performance after adaptation or training. Entries are mean$_{\pm \text{std}}$ success rates over 100 episodes $\times$ 3 seeds. Standard errors are at most $0.50/\sqrt{300} \approx 0.029$, so large drops such as color shift 25k$\rightarrow$ 100k are not explained by sampling noise.

Scenario	VLM-1k	VLM-5k	VLM-25k	VLM-100k	MAPPO trained	IPPO trained
Base scenario	$0.82_{\pm 0.38}$	$0.86_{\pm 0.48}$	$0.90_{\pm 0.30}$	$0.89_{\pm 0.32}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.00}$
Moving evaders	$0.99_{\pm 0.10}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.00}$
Larger grid (16×16)	$0.20_{\pm 0.40}$	$0.36_{\pm 0.48}$	$0.30_{\pm 0.46}$	$0.30_{\pm 0.46}$	$\mathbf{0.63}_{\pm 0.48}$	$0.27_{\pm 0.44}$
Additional evaders (2)	$0.52_{\pm 0.50}$	$\mathbf{0.67}_{\pm 0.47}$	$0.62_{\pm 0.48}$	$0.54_{\pm 0.46}$	$0.64_{\pm 0.48}$	$0.55_{\pm 0.50}$
Additional pursuers (3)	$0.63_{\pm 0.48}$	$0.58_{\pm 0.49}$	$0.80_{\pm 0.40}$	$0.60_{\pm 0.49}$	$\mathbf{1.00}_{\pm 0.00}$	$\mathbf{1.00}_{\pm 0.03}$
Color shift	$0.74_{\pm 0.44}$	$0.84_{\pm 0.37}$	$\mathbf{0.91}_{\pm 0.29}$	$0.56_{\pm 0.50}$	$0.63_{\pm 0.48}$	$0.00_{\pm 0.03}$
Semantic color swap	$0.58_{\pm 0.49}$	$0.42_{\pm 0.49}$	$0.59_{\pm 0.49}$	$\mathbf{0.61}_{\pm 0.49}$	$0.13_{\pm 0.34}$	$0.04_{\pm 0.18}$
Shape shift	$0.46_{\pm 0.50}$	$0.57_{\pm 0.49}$	$\mathbf{0.75}_{\pm 0.43}$	$0.64_{\pm 0.48}$	$0.15_{\pm 0.35}$	$0.13_{\pm 0.33}$
Non-centered ego	$0.10_{\pm 0.30}$	$0.17_{\pm 0.38}$	$\mathbf{0.22}_{\pm 0.41}$	$0.02_{\pm 0.13}$	$0.01_{\pm 0.07}$	$0.00_{\pm 0.03}$

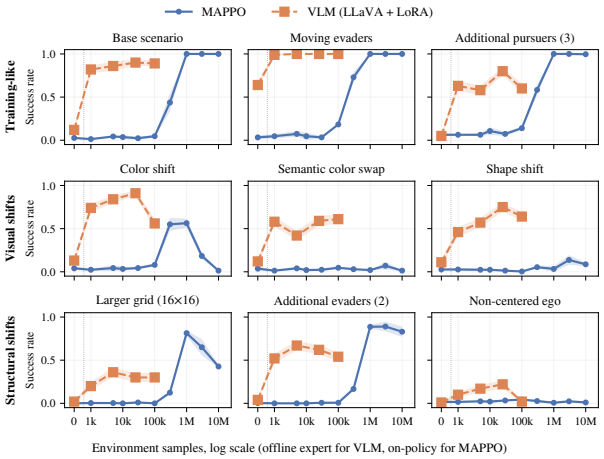


Figure 4. Per-shift learning curves comparing MAPPO with the adapted LLaVA-v1.6 + LoRA controller. The VLM axis denotes offline teacher demonstrations, while the MAPPO axis denotes on-policy environment samples. The axes are method-specific and are not unit-equivalent.

in-distribution LoRA fine-tuning specializes the visual representation to base-render statistics, partially eroding the backbone invariances that support transfer; frozen-vision-tower and domain-randomized controls are natural follow-up ablations.

5.4. Per-Shift Supervision Profiles

Figure 4 compares MAPPO and the adapted VLM across their own supervision axes: offline teacher demonstrations for the VLM and on-policy environment samples for MAPPO. These axes are not unit-equivalent and should not be read as matched-compute comparisons. The figure visualizes how each method changes as its natural form of supervision increases.

On training-like shifts, MAPPO eventually matches or exceeds the adapted VLM, especially on the base scenario and additional-pursuers setting. This reflects the strength of task-specialized on-policy optimization when deployment remains close to the training distribution. The adapted VLM reaches useful performance with fewer offline demonstrations on its own supervision axis, but those demonstrations come from a teacher whose own training cost is not included in the VLM budget.

On visual and semantic shifts, the adapted VLM remains stronger than MAPPO across most budgets. Since the teacher never sees these shifts, this gap is most naturally attributed to pretrained visual-semantic priors that survive distillation, rather than to the teacher policy alone. On structural shifts, the picture is mixed: MAPPO eventually does better on the larger grid, the two approaches are close with additional evaders, and both remain weak under non-centered egocentric observations.

Overall, the results support a calibrated division of strengths. MARL provides the strongest task-specialized performance when enough interaction is available and test conditions match training. VLM distillation provides a useful multimodal student that approaches teacher behavior on the base task and retains stronger visual-semantic robustness under some shifts, but it depends on a trained teacher and is not uniformly robust.

6. Discussion

The results should be interpreted through the student–teacher framing established in Section 3 and Section 4.2. The adapted VLM is distilled from a converged MAPPO teacher trained only on the base scenario, so the comparison is not between two independent learners. The "trained MARL" baseline in Table 2 is a separately trained MAPPO policy of the same family (Section A), not the exact distillation teacher checkpoint, but it serves as the task-specialized RL reference.

Within this framing, three observations stand out. First, zero-shot VLMs provide non-trivial behavior before task-specific supervision, while cold-start RL acts as a random-policy lower bound. This does not imply that VLMs are stronger controllers than RL in general; rather, it shows that pretrained multimodal representations provide usable initial

signal in this diagnostic pursuit setting. Second, the adapted VLM approaches teacher-level base-task performance with 25k–100k offline demonstrations, but this is data efficiency of distillation given a trained teacher, not an end-to-end replacement for MARL training. The teacher cost remains part of the overall pipeline.

Third, the adapted VLM is more robust than trained MARL under visual and semantic shifts that the teacher does not see during training. Since the teacher is trained only on the base scenario, this robustness is most plausibly attributed to pretrained visual–semantic priors that survive distillation, rather than to the teacher policy alone. Conversely, trained MAPPO remains strongest on the base task and some training-like or coordination-heavy variants, showing that task-specialized on-policy optimization remains more effective when deployment stays close to the training distribution. Overall, the results are best read as a decomposition of strengths: MARL supplies task-specialized behavior, the VLM backbone supplies visual–semantic invariances, and offline distillation combines the two imperfectly.

7. Limitations and Future Work

This study is a controlled diagnostic comparison rather than an exhaustive benchmark for multimodal multi-agent control. PettingZoo Pursuit is compact and synthetic, with an 8×8 base grid and a small discrete action space. This makes controlled shift analysis tractable, but the observed trends should be tested in environments with richer observations, more complex coordination structure, and longer-horizon objectives.

Several limitations follow from the student–teacher setup. The adapted VLM depends on a trained MAPPO teacher, so its distillation efficiency should not be interpreted as an independent alternative to MARL training. Offline demonstrations and on-policy environment samples are also different forms of supervision, and exact wall-clock times were not logged uniformly; matched-compute comparisons therefore remain future work. Additional ablations are also needed to isolate the source of visual robustness, including domain-randomized MAPPO, frozen-vision-tower LoRA, pure behavioral cloning versus AWR, demonstration-quality controls, and alternative prompt or action-decoding protocols. Finally, hybrid pipelines that warm-start MARL from adapted VLM policies may provide a more practical path than treating VLMs and MARL as separate controller families.

8. Conclusion

This paper presented a controlled student–teacher comparison of zero-shot VLMs, MAPPO-distilled VLMs, and PPO-style MARL baselines in partially observable cooperative pursuit. The adapted VLM is a multimodal student of a MAPPO teacher trained only on the base scenario, so its performance reflects both teacher-policy distillation and pretrained visual–semantic priors. Results show that zero-shot VLMs provide non-trivial behavior relative to random cold-start RL, that LoRA distillation with 25k–100k teacher demonstrations approaches teacher-level base-task performance, and that trained MAPPO remains strongest on the base task and some coordination-heavy variants. Under visual and semantic shifts unseen by the teacher, the adapted VLM is more robust, suggesting that pretrained VLM priors can survive distillation and support transfer beyond the teacher's training distribution. Both controller families remain weak under observation-layout shifts. These findings support evaluating multimodal multi-agent controllers along zero-shot utility, distillation efficiency, and source-of-robustness axes, rather than only final performance on the training task.

Acknowledgements

This material is based upon work supported by the National Science Foundation (NSF) under Grant No. 2434728. Any opinions, findings, and conclusions are those of the authors and do not necessarily reflect the views of the NSF.

LLM/Agent Usage Disclosure

The authors conducted the core research activities for this work, including problem formulation, experiment design, implementation, debugging, analysis, and interpretation of results. Language-model assistance was used as a supporting tool for drafting, editing, restructuring text, and providing coding support. All experiments, results, and scientific claims were developed, checked, and validated by the authors.

References

Ahn, M., Brohan, A., Brown, N., Chowdhery, N., Herzog, A., Ibarz, J., Lee, D., Lynch, C., McGrew, J., Nair, A., Xu, P., Levine, S., et al. Do as i can, not as i say: Grounding language in robotic affordances. In *Proceedings of Robotics: Science and Systems (RSS)*, 2022. URL https://arxiv.org/abs/2204.01691.

Amato, C. A first introduction to cooperative multi-agent reinforcement learning. *arXiv preprint arXiv:2405.06161*, 2024. URL https://arxiv.org/abs/2405.06161.

Bai, J., Bai, S., Yang, S., Wang, S., Tan, S., Wang, P., Lin, J., Zhou, C., and Zhou, J. Qwen-vl: A versatile vision-language model for understanding, localization, text read-

ing, and beyond. *arXiv preprint arXiv:2308.12966*, 2023. URL https://arxiv.org/abs/2308.12966.

de Witt, C. S., Gupta, T., Makoviichuk, D., Makoviychuk, V., Torr, P. H. S., Sun, M., and Whiteson, S. Is independent learning all you need in the StarCraft multi-agent challenge? *arXiv preprint arXiv:2011.09533*, 2020. URL https://arxiv.org/abs/2011.09533.

Dettmers, T., Pagnoni, A., Holtzman, A., and Zettlemoyer, L. QLoRA: Efficient finetuning of quantized LLMs. In *Advances in Neural Information Processing Systems*, volume 36, pp. 10088–10115, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/1feb87871436031bdc0f2beaa62a049b-Abstract-Conference.html.

Driess, D., Xia, F., Sajjadi, M. S. M., Lynch, C., Chowdhery, A., Ichter, B., Wahid, A., Tompson, J., Vuong, Q., Yu, T., Huang, W., Chebotar, Y., Sermanet, P., Duckworth, D., Levine, S., Vanhoucke, V., Hausman, K., Toussaint, M., Greff, K., Zeng, A., Mordatch, I., and Florence, P. Palm-e: An embodied multimodal language model. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 8469–8488, 2023. URL https://proceedings.mlr.press/v202/driess23a.html.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. LoRA: Low-rank adaptation of large language models. In *Proceedings of the 10th International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Jiang, Y., Gupta, A., Zhang, Z., Wang, G., Dou, Y., Chen, Y., Fei-Fei, L., Anandkumar, A., Zhu, Y., and Fan, L. Vima: Robot manipulation with multimodal prompts. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pp. 14991–15019, 2023. URL https://proceedings.mlr.press/v202/jiang23b.html.

Liu, H., Li, C., Wu, Q., and Lee, Y. J. Visual instruction tuning. In *Advances in Neural Information Processing Systems*, volume 36, pp. 34892–34916, 2023. URL https://proceedings.neurips.cc/paper_files/paper/2023/hash/6dcf277ea32ce3288914faf369fe6de0-Abstract-Conference.html.

Oliehoek, F. A. and Amato, C. *A Concise Introduction to Decentralized POMDPs*. Springer, 2016. doi: 10.1007/978-3-319-28929-8. URL https://link.springer.com/book/10.1007/978-3-319-28929-8.

Peng, X. B., Kumar, A., Zhang, G., and Levine, S. Advantage-weighted regression: Simple and scalable off-policy reinforcement learning. *arXiv preprint arXiv:1910.00177*, 2019. URL https://arxiv.org/abs/1910.00177.

Terry, J. K., Grammel, N., Hari, A., Santos, L., Perez-Vicente, R. D. L., Jiang, A. G. H., Santos, R. O., Silva, R. H., Blackburn, K., et al. Pettingzoo: Gym for multi-agent reinforcement learning. *arXiv preprint arXiv:2009.14471*, 2021. URL https://arxiv.org/abs/2009.14471.

Yu, C., Velu, A., Vinitsky, E., Gao, J., Wang, Y., Bayen, A., and Wu, Y. The surprising effectiveness of ppo in cooperative multi-agent games. In *Advances in Neural Information Processing Systems*, volume 35, pp. 24611–24624, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/9c1535a02f0ce079433344e14d910597-Abstract-Datasets_and_Benchmarks.html.

Zitkovich, B., Yu, T., Xu, S., Xu, P., Xiao, T., Xia, F., Wu, J., Wohlhart, P., Welker, S., Wahid, A., Vuong, Q., Vanhoucke, V., Tran, H., Soricut, R., Singh, A., Singh, J., Sermanet, P., Sanketi, P. R., Salazar, G., Ryoo, M. S., Reymann, K., Rao, K., Pertsch, K., Mordatch, I., Michalewski, H., Lu, Y., Levine, S., Lee, L., Lee, T.-W. E., Leal, I., Kuang, Y., Kalashnikov, D., Julian, R., Joshi, N. J., Irpan, A., Ichter, B., Hsu, J., Herzog, A., Hausman, K., Gopalakrishnan, K., Fu, C., Florence, P., Finn, C., Dubey, K. A., Driess, D., Ding, T., Choromanski, K. M., Chen, X., Chebotar, Y., Carbajal, J., Brown, N., Brohan, A., Arenas, M. G., and Han, K. Rt-2: Vision-language-action models transfer web knowledge to robotic control. In *Proceedings of The 7th Conference on Robot Learning*, volume 229 of *Proceedings of Machine Learning Research*, pp. 2165–2183, 2023. URL https://proceedings.mlr.press/v229/zitkovich23a.html.

Appendix

A. Training Details

This appendix reports the environment settings, training configurations, inference protocol, uncertainty calculation, and compute context used for the experiments in the main text. Together with Section 4.2 and Section 4.3, these details are intended to make the evaluation reproducible.

A.1. Environment Configuration

All pursuit experiments use `pettingzoo.sisl.pursuit_v4`. The base configuration uses 2 pursuers, 1 evader, an 8×8 grid, $n_$catch $= 2$, `surround=False`, per-agent non-shared rewards, and `max_cycles`$= 100$. The base scenario uses frozen evaders; the *moving evaders* shift replaces them with random-walking evaders. The *larger grid* shift uses a 16×16 grid, and the *additional evaders* and *additional pursuers* shifts increase the corresponding agent count while keeping other parameters fixed. Observation and action spaces are padded across agents using SuperSuit.

The MAPPO teacher used to generate the offline VLM-distillation dataset is trained only on the base configuration, without domain randomization or exposure to any visual, semantic, or structural shift used at evaluation time. This is why cross-shift robustness of the adapted VLM is interpreted in the main text as arising from pretrained VLM priors, coverage in the base-scenario demonstrations, or their interaction, rather than from teacher exposure to those shifts.

A.2. MAPPO and IPPO Training

Two MAPPO configurations appear in this paper.

Distillation teacher. The teacher used to label offline demonstrations is trained with `scripts/train/train_pursuit_mappo.py` and acts on the raw $7 \times 7 \times 3$ PettingZoo tensor observation. Its policy outputs are stored as teacher actions in the JSONL records used for VLM distillation. The rendered RGB images stored alongside those records are produced by a separate render pass and are the inputs used by the VLM student.

Obs-image MAPPO/IPPO baselines. The "MAPPO trained" and "IPPO trained" columns in Table 2, as well as the MAPPO learning curves in Figure 4, use policies trained with `train_pursuit_mappo_obs_images_dense_ckpt.py` and `train_pursuit_ippo_obs_images.py`. These baselines act on rendered RGB observations produced by the training-time `ObservationImageWrapper`. They are therefore matched to the VLMs in local information content, but not in exact pixel rendering.

Both variants use RLlib PPO with a custom `RLModule`. MAPPO uses a centralized value head over concatenated agent observations and shares one policy across pursuers. IPPO uses independent critics and one independent policy per pursuer. The teacher and obs-image baselines differ in observation interface and network trunk, but otherwise use the same PPO hyperparameters. Values labeled "RLlib default" inherit RLlib's PPO new-API defaults.

Distillation teacher checkpoint. All VLM training budgets use the same converged MAPPO teacher checkpoint, denoted `checkpoint_000079`. This checkpoint comes from the MAPPO run dated `2025-09-02` and is reused for the 1k, 5k, 25k, and 100k demonstration budgets. It also supplies the fixed 5k held-out validation set used in Figure 3.

A.3. Distillation Data Collection

Offline demonstrations are collected from the MAPPO teacher checkpoint above using the distillation-data collection script `scripts/data/collect_mappo_distill_data_with_values.py`. For each transition, the script records the teacher action a_t, centralized-critic value $V(s_t)$, action logits, immediate reward r_t, return-to-go G_t, one-step Q-estimate $Q(s_t, a_t) = r_t + \gamma V(s_{t+1})$, and advantage A_t. All reported VLM runs use the one-step TD advantage

$$A_t = r_t + \gamma V(s_{t+1}) - V(s_t), \quad \gamma = 0.99.$$

Each local observation is rendered to RGB using `obs_to_image`. The renderer colorizes walls as black, the ego pursuer as blue, allied pursuers as green, and evaders as red, and overlays per-cell pursuer/evader counts. The render scale is

$$\texttt{scale} = \max(16, \lfloor 256 / \max(H, W) \rfloor),$$

Table 3. PPO hyperparameters for the MAPPO distillation teacher and the obs-image MAPPO/IPPO baselines.

Hyperparameter	Distillation teacher	Obs-image MAPPO/IPPO
Algorithm	PPO (RLlib new API)	PPO (RLlib new API)
Framework	PyTorch	PyTorch
Training script	`train_pursuit_mappo.py`	`*_obs_images*.py`
Policy sharing	Shared across pursuers	Shared (MAPPO) / independent (IPPO)
Critic	Centralized	Centralized (MAPPO) / independent (IPPO)
Observation	Raw $7\times7\times3$ tensor	RGB image, $112\times112\times3$
Actor/critic trunk	CNN $[32, 64, 128]$ + FC $[128, 128]$, ReLU	CNN $[32, 64, 128]$ + FC $[256, 256]$, ReLU
Rollout workers	5	4
Envs per worker	8	2
Train batch size	512	512
Minibatch size	64	64
SGD epochs/update	10	10
Total env. samples	teacher checkpoint 79	up to 10^7
Learning rate	RLlib default (5×10^{-5})	RLlib default (5×10^{-5})
Discount γ	RLlib default (0.99)	RLlib default (0.99)
GAE λ	RLlib default (1.0)	RLlib default (1.0)
PPO clip ϵ	RLlib default (0.3)	RLlib default (0.3)
Entropy coefficient	RLlib default (0.0)	RLlib default (0.0)
VF clip	RLlib default (10.0)	RLlib default (10.0)
Evaluation seeds	—	$42, 123, 456$

which gives 36 pixels per cell for the default 7×7 local observation, producing a $252 \times 252 \times 3$ image. This differs from the obs-image RL baseline renderer, which uses 16 pixels per cell and no count overlay. Thus, the comparison is matched in local state information, not in exact pixel encoding.

A.4. VLM Fine-Tuning

The VLM controller is initialized from `llava-hf/llava-v1.6-mistral-7b-hf` and fine-tuned with LoRA adapters on top of a 4-bit NF4 QLoRA-quantized base model. Each training example contains a rendered local observation image, task prompt, teacher action, and value/advantage annotations. Training budgets of 1k, 5k, 25k, and 100k demonstrations are evaluated. For each budget, the reported model is the best validation-checkpoint model selected by `eval_loss` using Hugging Face Trainer's `load_best_model_at_end`. Validation uses a fixed 5k held-out demonstration set sampled from the same teacher distribution.

A.5. Inference and Evaluation

At evaluation time, VLM and RL controllers act greedily from local observations. The adapted LLaVA-v1.6 controller uses the constrained logit-scoring protocol in Section 4.3 with `max_new_tokens=2` and `do_sample=False`. Zero-shot LLaVA-v1.6 and Qwen-VL use deterministic free generation followed by the regex parser described in Section 4.3; parse failures default to "stay" and are logged as invalid actions. MAPPO and IPPO controllers act greedily from loaded checkpoints through the RLlib inference path.

Each scenario is evaluated over 100 episodes per seed and 3 random environment seeds, for $N = 300$ trials per cell. Episodes terminate at `max_cycles`$= 100$ or when all evaders are captured.

A.6. Uncertainty Calculation

Main tables report success rates as mean$_{\pm\text{std}}$ over $N = 300$ binary trials. The reported standard deviation is the per-trial binary standard deviation. The standard error of each mean is

$$\text{SE} = \frac{\text{std}}{\sqrt{300}}.$$

Table 4. LLaVA-v1.6 LoRA fine-tuning hyperparameters for advantage-weighted regression. Effective batch size is $6 \times 2 = 12$.

Hyperparameter	Value
Backbone	`llava-hf/llava-v1.6-mistral-7b-hf`
Quantization	QLoRA, NF4, double-quant, bf16 compute
Adaptation	LoRA
LoRA rank r	16
LoRA α	32
LoRA dropout	0.05
LoRA bias	none
Target modules	`q,k,v,o,gate,up,down proj.`
Task type	`CAUSAL_LM`
Optimizer	AdamW
Learning rate	5×10^{-5}
LR schedule	cosine, warmup ratio 0.1
Weight decay	0.01
Max gradient norm	1.0
Per-device batch size	6
Gradient accumulation	2
Effective batch size	12
Max sequence length	1024 tokens
Epochs	20 with early stopping
Early-stopping metric	`val_loss`
Early-stopping patience	12 eval steps
Early-stopping min. delta	0.003
Early-stopping warmup	3 epochs
Precision	bf16, TF32 enabled
Gradient checkpointing	on
AWR advantage type	one-step TD
AWR discount γ	0.99
AWR advantage clip	2.5
AWR temperature β	1.0
AWR normalization	per-batch
Auxiliary value-loss coefficient	0.5
Training budgets	1k, 5k, 25k, 100k
Validation set	fixed 5k held-out demonstrations
Training seed	42
Evaluation seeds	42, 123, 456

Since the largest possible binary standard deviation is 0.5, all reported means have $\mathrm{SE} \leq 0.5/\sqrt{300} \approx 0.029$. The standard error of a difference between two independent means is therefore at most

$$\mathrm{SE}_\Delta = \sqrt{2} \cdot \frac{0.5}{\sqrt{300}} \approx 0.041.$$

This bound is used in the main text to interpret the large non-monotonic drops from 25k to 100k demonstrations on color shift and non-centered ego. The full per-cell standard deviations are already included directly in Tables 1 and 2; therefore, a duplicate appendix standard-deviation table is omitted.

A.7. Compute

VLM fine-tuning runs were executed on SLURM jobs with 8× NVIDIA L40 GPUs. MAPPO/IPPO training runs used a single GPU on the same partition with rollout workers as listed in Table 3. Exact wall-clock times were not logged uniformly across VLM fine-tuning and MARL training, so matched-compute comparisons are not reported. Accordingly, the main text treats offline VLM demonstrations and on-policy MARL environment samples as method-specific supervision axes rather than unit-equivalent compute budgets.