

Reward Design for Vision-Assisted Autonomous Driving in MetaDrive: From Scalar Rewards to Event-Triggered Visual Guidance

Zhenhong Guo

*Southern University of Science and Technology, Shenzhen, China
2519416535@qq.com*

Abstract. Visual feedback can characterize traffic interactions, yet a plausible visual score may not promote task completion. This paper examines this mismatch in a MetaDrive intersection task governed by state-based Proximal Policy Optimization (PPO). With matched 1.8-million-step training, a distilled scalar visual reward raises return but reduces success from 0.388 to 0.287. Event-Triggered Visual Guidance (ETVG) instead uses a frozen ResNet18 to process one overlaid top-down RGB frame every five environment steps and determine whether the ego vehicle should yield. At the first qualifying event, ETVG preserves PPO steering and caps only the longitudinal action for five steps. Offline labels derive from path conflicts and relative arrival times; deployment uses only current RGB. The frozen visual layer is attached to three independently trained PPO policies, each evaluated on 300 matched scenarios. Mean success increases from 0.338 to 0.410, with a significant paired gain for every policy seed ($p \leq 0.0201$); collision and road-departure rates also decrease. On an episode-disjoint test, the detector reaches 0.9849 ROC AUC and 0.8337 average precision. Exact-reset, state-trigger, raw-image, traffic-density, joint-training, and local-VLM ablations indicate that the gain arises from binding a single visual event to a restricted action rather than accumulating a general visual score.

Keywords: autonomous driving, reward design, computer vision, reinforcement learning, MetaDrive

1. Introduction

An autonomous-driving policy is evaluated by route completion without collision or road departure, not by accumulated training reward. Simulation makes failures repeatable but does not solve objective specification: an agent still learns the implemented reward, including any unintended shortcuts. MetaDrive provides controllable seeds and compositional roads [1], while CARLA and AirSim provide complementary urban, visual, and physical settings [2, 3]. Across these simulators, reward specification and evaluation remain central to reinforcement-learning-based driving [4].

This mismatch is particularly evident at intersections. Progress and speed can encourage a vehicle to enter an occupied conflict area, whereas a large generic penalty can produce hesitation without teaching when to yield. An image may reveal an interaction that is difficult to capture with a few

state thresholds, but a visual score is useful only if its action improves completion. Conditional imitation learning maps images and navigation commands directly to control [5], and ROAD distills VLM and traffic-rule judgments for post-collision responsibility [6]. This study asks a narrower question: can current RGB identify a yielding event before contact and support a bounded correction to an otherwise state-based policy?

We first compare baseline, terminal-safety, comfort, multi-objective, and scalar visual reward formulations under matched evaluation. We retain the failed scalar visual approach: after matched 1.8-million-step training, it raises shaped return but reduces success. This failure motivates Event-Triggered Visual Guidance (ETVG), which replaces an accumulated visual score with a single RGB-derived yielding event and a brief longitudinal cap. The detector is trained offline on path-conflict and relative-arrival labels, whereas deployment uses pixels rather than those privileged quantities.

The main contribution is an auditable event-action interface rather than a new end-to-end driver. A frozen detector and rule are attached to three independently trained PPO policies, each evaluated on the same 300 held-out scenarios as its control. Exact-reset branches isolate the tested action, a privileged state trigger checks whether RGB contributes useful timing information, and frozen-rule sensitivity and traffic-density tests probe robustness. Local-VLM and joint-training experiments provide negative or boundary evidence. This design analyzes all routes rather than selecting only outcomes that support the final method.

2. Related work

2.1. Simulation and learning-based driving

MetaDrive exposes deterministic scenario identifiers, configurable traffic, and compositional road networks [1]. CARLA emphasizes urban photorealism [2], while AirSim provides high-fidelity visual and physical simulation [3]. These platforms support end-to-end vision, imitation, and reinforcement learning, but they do not eliminate the need to control randomness during training and evaluation. Conditional imitation learning shows that images can support driving when paired with route commands [5]. In the present study, the controller remains an MLP over MetaDrive state features so changes can be attributed to rewards or the visual interface rather than a changed policy architecture.

Accordingly, PPO [7] is held fixed throughout the main comparison. SAC [8] is a relevant continuous-control alternative, but introducing it would add optimizer and action-distribution factors to an already multifactorial reward study. The goal is not to rank algorithms but to determine how visual information should enter a controlled PPO pipeline.

2.2. Reward shaping and visual feedback

Driving rewards commonly mix progress, speed, terminal outcomes, lane position, time-to-collision, and comfort. Potential-based shaping can preserve an optimal policy under specific assumptions [9], but practical driving terms seldom satisfy these conditions. A higher return can therefore coexist with a lower task-success rate. ROAD uses responsibility-aware visual and traffic-rule judgments to scale crash penalties [6]; ETVG instead detects a pre-collision yielding event and limits one action dimension before contact.

Vision-language models can provide semantic rewards [10], pairwise preference feedback [11], and multimodal instruction following [12]. Their broad semantics help when an event is hard to specify, but frequent inference is slow, and plausible verbal judgments may not yield effective

actions. An ImageNet-pretrained ResNet18 [13, 14] offers a smaller, testable detector for a single event. Action shielding and invalid-action masking motivate separating policy proposals from restricted execution [15, 16]. ETVG adopts this separation but does not claim a formal safety guarantee: the visual layer can impose only one brief longitudinal cap, and it cannot change steering or route choice.

3. Methodology

3.1. Task, PPO, and reward variants

The task is a finite-horizon Markov decision process. At each step t , the source policy receives the default 259-dimensional state o_t and selects $a_t=(i_p, l_p)$ from ten steering and ten longitudinal bins. Episodes terminate upon success, collision, road departure, or 500 steps. The discrete indices are converted by MetaDrive to normalized steering and longitudinal controls.

PPO uses $lr=3e-4$, a rollout size of 4,096, a batch size of 256, and $\gamma=0.99$. Its clipped surrogate objective follows [7]: $L_{clip}(\theta)=E_t[\min(r_t(\theta)A_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)A_t)]$, where $r_t(\theta)$ is the updated-policy-to-behavior-policy likelihood ratio, not the environment reward, and A_t is an advantage estimate. The controller remains a state-based MLP. Vision is introduced only through the tested scalar reward or event interface, so the source action distribution and route-following mechanism remain comparable.

MetaDrive combines progress and speed with a success reward of +40 and collision/out-of-road penalties of -20. Table 1 summarizes the implemented additions. A runtime audit found neutral fallbacks (0 m, 50 m, and 10 s) for unavailable lane-offset, distance, and TTC fields. These branches evaluate to zero and are excluded from component claims. Although artifact names retain "Safety," the executed Safety term is limited to $r_{term}=-5I_{crash}-5I_{out}$. Comfort uses $s_t=|v_t-v_{t-1}|$, an adjacent-step speed-change proxy rather than calibrated physical jerk. MultiReward combines the active terminal, speed-change, progress, and target-speed terms; it is not an exhaustive weight search.

Table 1. Implemented state-reward variants

Variant	Enabled additions	Exact settings
Baseline	None	Driving 1.0; speed 0.1; success +40; crash/out -20
Safety (terminal)	Terminal safety	Additional crash/out -5
Comfort	Speed-change proxy	Threshold 0.2; coefficient -0.1; scale 1.0; excess ratio capped at 3.0
MultiReward-v1	Safety, comfort, efficiency	Crash/out -5; progress 0.05; target speed 8 m/s; speed score 0.02
MultiReward-v2	Retuned safety, comfort, efficiency	Crash/out -6; comfort threshold 0.4, coefficient -0.03; progress 0.03; speed score 0.005

3.2. Why the scalar visual reward was rejected

The initial visual approach distilled VLM-assisted weak labels into a scalar visual reward: $r_{total}=r_{MD}+r_{term}+\alpha r_{visual}$, with $\alpha=0.02$. For a distilled score q_p , $r_{visual}=clip(\min(q_p+0.3, 0), -0.1, 0)$ and magnitudes below 0.01 become zero. The branch is non-positive and does not clip the native or terminal rewards.

This small-magnitude term accumulates over many process steps, whereas route success is terminal. A policy can reduce visual penalties by moving slowly or avoiding difficult interactions without completing the route. The matched 1.8-million-step experiment in Section 5 demonstrates this pattern: shaped return rises while success falls. The scalar approach is retained in the paper as an alignment diagnostic, not as a component of the supported controller.

3.3. RGB observations, offline labels, and detector

ETVG observes an 800x800 top-down RGB image with overlays. The ego vehicle is green; route, traffic paths, speed, and position markers are visible. The frame contains neither future outcomes nor hand-drawn danger labels and is not a front-camera view.

For offline supervision, planned paths are sampled every 2 m for up to 70 m, and traffic within 65 m is considered. A conflict requires paths within 3 m at a crossing angle of at least 35 degrees. Arrival times are $\tau_e = d_e / \max(v_e, 2)$ and $\tau_o = d_o / \max(v_o, 2)$. A positive target requires both times to be at most 2 s and $-0.25 \leq \tau_e - \tau_o \leq 0.25$ s. Future episode outcomes are unused. Conflict distances and arrival times are not numerical deployment inputs; online rendering includes planned paths and current position/speed markers. Thus, offline labels use path geometry and arrival times, whereas the deployed detector receives pixels only.

The dataset contains 800 complete episodes, 400 from each of two PPO checkpoints. The resulting 15,657 records include 800 terminal frames reserved for coverage and outcome auditing. Process frames are split by episode rather than image to prevent adjacent-frame leakage: 10,401/2,217/2,239 training/validation/test frames with 893/190/191 positives. The positive share is therefore about 8.6%, making conventional accuracy uninformative alone.

Images are resized to 224x224 and normalized with ImageNet statistics [14]. In a pretrained ResNet18 [13], the stem through layer2 is frozen; layer3, layer4, and a one-logit head are trained. A weighted sampler handles class imbalance. BCE-with-logits and AdamW use $\text{lr}=1\text{e-}4$, weight decay $1\text{e-}4$, batch size 32, seed 1947, a 30-epoch limit, and patience 6. Validation ROC AUC selects epoch 15 for frozen TorchScript export. For image I , $p_{\text{yield}}(t) = \text{sigmoid}(f_{\text{phi}}(I))$. This probability replaces the earlier four-dimensional driving-quality score and indicates only whether a brief yield is justified. Detector calibration is reported as a diagnostic, but no separate post-hoc calibration model is fitted.

3.4. Event-triggered visual guidance

Figure 1 separates ETVG's offline label construction, online visual intervention, and terminal-frame audit.

The online intervention is limited. Every five steps, one current RGB frame is processed, about once per 0.5 s. A candidate requires $p_{\text{yield}}(t) \geq 0.60$ and a PPO longitudinal index of at least 6. With ten bins, index j maps to $2j/9-1$, so the gate requires at least +0.333 acceleration. Only the first candidate event in an episode is accepted. During the next five steps, steering is preserved exactly and $l_{\text{exec}}(t+k) = \min(l_{\text{PPO}}(t+k), 3)$, $k=1, \dots, 5$; cap index 3 maps to -0.333 and does not weaken stronger PPO braking. Later detections are logged but trigger no further intervention. ETVG cannot select a route, change steering, or repeatedly stop the vehicle.

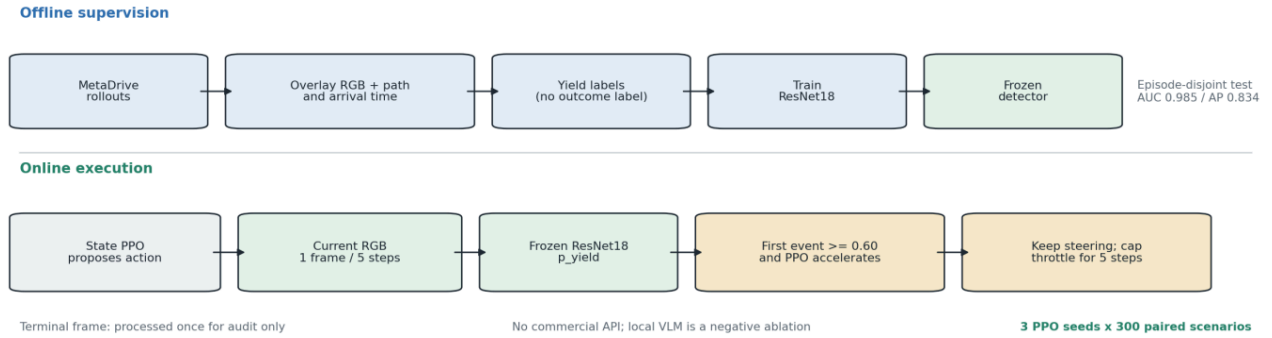


Figure 1. ETVG pipeline. Offline supervision, online ETVG execution, and terminal auditing are separated. Path conflict and arrival time create labels; deployment uses current RGB and a bounded action rule

The actual final frame is always processed exactly once. If termination coincides with a periodic step, only the terminal branch is counted. This audit does not alter any preceding actions. The supported controller uses neither a VLM nor a commercial API online.

3.5. Evaluation protocol

Success is primary; collision, out-of-road, route completion, and return explain the changes. Matched scenarios report W control failures rescued by ETVG and L control successes harmed by ETVG, with an exact two-sided binomial test over W+L. Exact-reset action ablations use POSIX fork so simulator, traffic, policy, and random-number states match before the tested action. Parameters are frozen before sensitivity tests. A replacement requires a success gain of at least +0.02 and a paired $p < 0.05$, no higher collision or road-departure rates, and confirmation on new seeds.

Each PPO seed is tested separately because scenario identifiers recur across policy lineages. The pooled rescue/harm count summarizes episode-level behavior but does not constitute an independent test of training randomness. Exact-reset branches begin from the same post-reset process state; the first intentional difference is the tested longitudinal rule. Terminal coverage, rendering failures, and steering modifications are audited alongside the outcome metrics.

4. Experimental setup

4.1. Software, hardware, and scenario

The implementation uses Python 3.10.20, MetaDrive 0.4.3, Gymnasium 1.3.0, Stable-Baselines3 2.9.0 [17], and PyTorch 2.7.1+cu118. PPO runs on Intel Xeon E5-2680 v4 CPUs; ResNet training and evaluation use TITAN V 12-GiB GPUs. VLM diagnostics run locally. The supported controller calls neither a VLM nor a commercial API.

The primary task uses intersection map="X", density 0.5, random spawn lanes, 2,000 training scenarios from seed 5, and a 500-step horizon. Traffic-free straight and curved maps are engineering checks only. Density 0.8 provides a bounded traffic-load shift. The roundabout is excluded because source-policy success is 0/100; evaluating an intervention on a policy that never completes the target geometry would not isolate the visual contribution.

4.2. Policy training and compared methods

Three independent policy lineages use PPO seeds 0, 1, and 2. Each trains terminal Safety for 1.8 million requested steps; 4,096-step rollouts yield 1,802,240 actual steps. Checkpoints are saved every 100,000 steps. No custom policy_kwarg are supplied, so Stable-Baselines3 uses its default MlpPolicy; state-reward runs add no global reward normalization or clipping.

Each lineage then receives a 100,000-step continuation without visual input. Its checkpoint is evaluated twice on identical scenario seeds: unchanged Control and the frozen interval5/threshold0.60/first-event/cap3/hold5 ETVG wrapper. Policy weights are therefore matched within each control-visual pair, and the reported gain cannot be attributed to additional visual policy training.

The main evaluation uses 300 unseen seeds 17000-17299 per lineage, giving 900 matched control-visual pairs. Non-overlapping blocks test scalar reward (12600-13199), RGB versus privileged state triggers (16700-16999), sensitivity (18300-18599), density 0.8 (18600-18899), and state rewards (19100-19399). Compared methods include native PPO, terminal Safety, Comfort, MultiReward-v1/v2, scalar visual reward, frozen Control, ETVG, a privileged State Trigger with the same action, raw-image input, and local Qwen2.5-VL-7B diagnostics. The State Trigger is an information-rich reference, not a deployable baseline.

5. Results

5.1. Reward shaping and scalar-reward misalignment

Table 2 combines two matched protocols. The first five rows evaluate independently trained state-reward policies on seeds 19100-19399, with 300 episodes per method; the last three evaluate from-scratch 1.8-million-step policies on 600 common seeds. The protocols remain separate.

Table 2. State-reward and scalar visual-reward diagnostics

Policy	Success	Crash	Out	Route	Return
Baseline	0.363	0.550	0.087	0.726	45.193
Safety (terminal)	0.373	0.600	0.027	0.722	43.705
Comfort	0.277	0.530	0.193	0.708	47.144
MultiReward-v1	0.153	0.653	0.193	0.626	31.703
MultiReward-v2	0.243	0.653	0.103	0.694	28.871
Baseline, from scratch	0.350	0.573	0.077	0.721	40.773
Safety, from scratch	0.388	0.588	0.023	0.724	45.496
Safety + scalar visual reward	0.287	0.617	0.097	0.698	50.692

The first five rows use the same 300 non-wrapping seeds 19100-19399; all timeouts are zero. Safety changes success by only +0.010 ($p=0.832$): road departures fall from 0.087 to 0.027, but collisions rise from 0.550 to 0.600. Comfort improves the speed-change proxy from 0.229 to 0.219 while reducing success and increasing road departures. Both MultiReward settings are significantly worse than Baseline. Mean per-step additions confirm the active terms: Safety -0.0327, Comfort -0.0040, and MultiReward-v1/v2 terminal safety (-0.0521/-0.0573), comfort (-0.0051/-0.0008), and

efficiency (+0.0373/+0.0159). The two MultiReward settings serve as fixed comparisons rather than a global weight search.

The last three rows use a separate matched 600-seed from-scratch protocol. Relative to Safety, the scalar visual reward produces 90 rescues and 151 harms ($p=0.000103$); success falls by 0.102 with bootstrap 95% CI $[-0.152, -0.052]$, despite the highest return. This is not a minor fluctuation around equal performance: PPO learned the dense proxy while completing fewer routes.

5.2. Detector and main ETVG result

On 2,239 episode-disjoint process frames, including 191 positives, ResNet18 achieves 0.9849 ROC AUC, 0.8337 average precision, a 0.0327 Brier score, and 0.0299 expected calibration error. A precision-oriented rule initially selected 0.95, yielding test precision/recall 0.824/0.613, but failed the recall gate. Threshold 0.60 was then fixed for higher recall before the final policy seed blocks; test precision, recall, and specificity are 0.780, 0.743, and 0.980. The final downstream outcomes were not used to retune this threshold.

With split, optimizer, and seed fixed, classifier-only and layer4+fc adaptations achieve AUC/AP 0.9584/0.6220 and 0.9821/0.8093. The deployed layer3+layer4+fc model yields the best average precision and calibration. These metrics show that the event is learnable from the overlay distribution, but not that acting on it improves driving. Table 3 therefore reports matched controller results for three independent PPO lineages.

Table 3. Frozen ETVG replication across independent PPO training seeds (300 matched scenarios each)

PPO seed	Control	ETVG	Delta	Crash C/V	Out C/V	W/L; exact p
0	0.343	0.410	+0.067	0.597/0.550	0.060/0.040	30/10; 0.002221
1	0.250	0.353	+0.103	0.500/0.437	0.250/0.210	41/10; 0.000015
2	0.420	0.467	+0.047	0.400/0.367	0.180/0.167	23/9; 0.020062
Mean	0.338	0.410	+0.072	0.499/0.451	0.163/0.139	94/29; descriptive

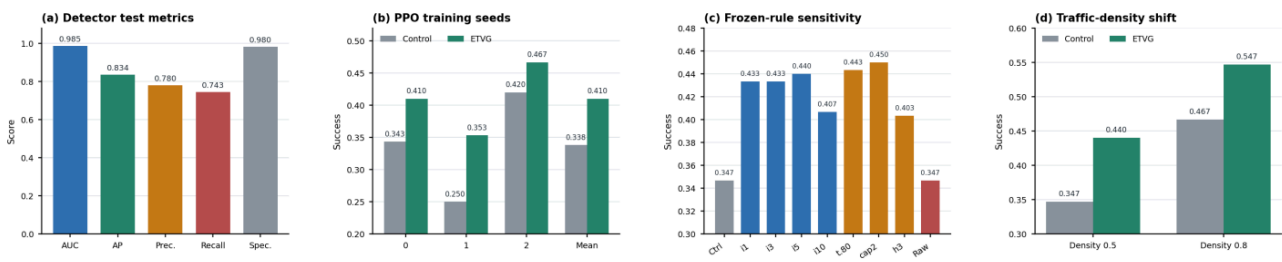


Figure 2. Held-out detector metrics, three PPO training seeds, frozen-rule sensitivity, and a same-geometry traffic-density shift

Each lineage exhibits a positive paired improvement ($p \leq 0.0201$). Mean collision and out-of-road rates fall by 0.048 and 0.024, respectively, while route completion rises from 0.727 to 0.753. The accepted-event fraction ranges from 0.377 to 0.560. Terminal coverage is complete, render errors are zero, and no steering modifications occur. Because scenario IDs recur across lineages, the pooled 94/29 count and p-value are descriptive; evidence across training randomness comes from

the three policy-level rows. Figure 2 summarizes detector quality, policy-seed replication, frozen-rule sensitivity, and the traffic-density shift.

5.3. Ablations, robustness, and runtime

On exact-reset seeds 18300-18599, main ETVG raises success from 0.347 to 0.440 (W/L=33/5, $p=4.26e-6$) and reduces collision/out-of-road rates from 0.600/0.053 to 0.527/0.033. Intervals 1/3/5/10 yield 0.433/0.433/0.440/0.407; every-step inference offers no gain. Thresholds 0.50-0.80 yield 0.440-0.443. Caps 4/3/2 yield 0.410/0.440/0.450, but cap 2 is not significantly better ($p=0.629$). Hold durations 3/5/8 yield 0.403/0.440/0.423. No candidate satisfies the replacement rule.

With cap3/hold5 fixed, the Control, RGB, and privileged State-trigger variants achieve 0.317/0.420/0.427 success on 300 exact-reset seeds. State versus RGB yields 10 wins and 8 losses ($p=0.814529$), indicating that RGB matches complete path-conflict and arrival-time information. Applying overlay-trained weights to raw RGB reduces the event rate from 0.513 to 0.053 and returns success to the control value (0.347), showing input-distribution dependence.

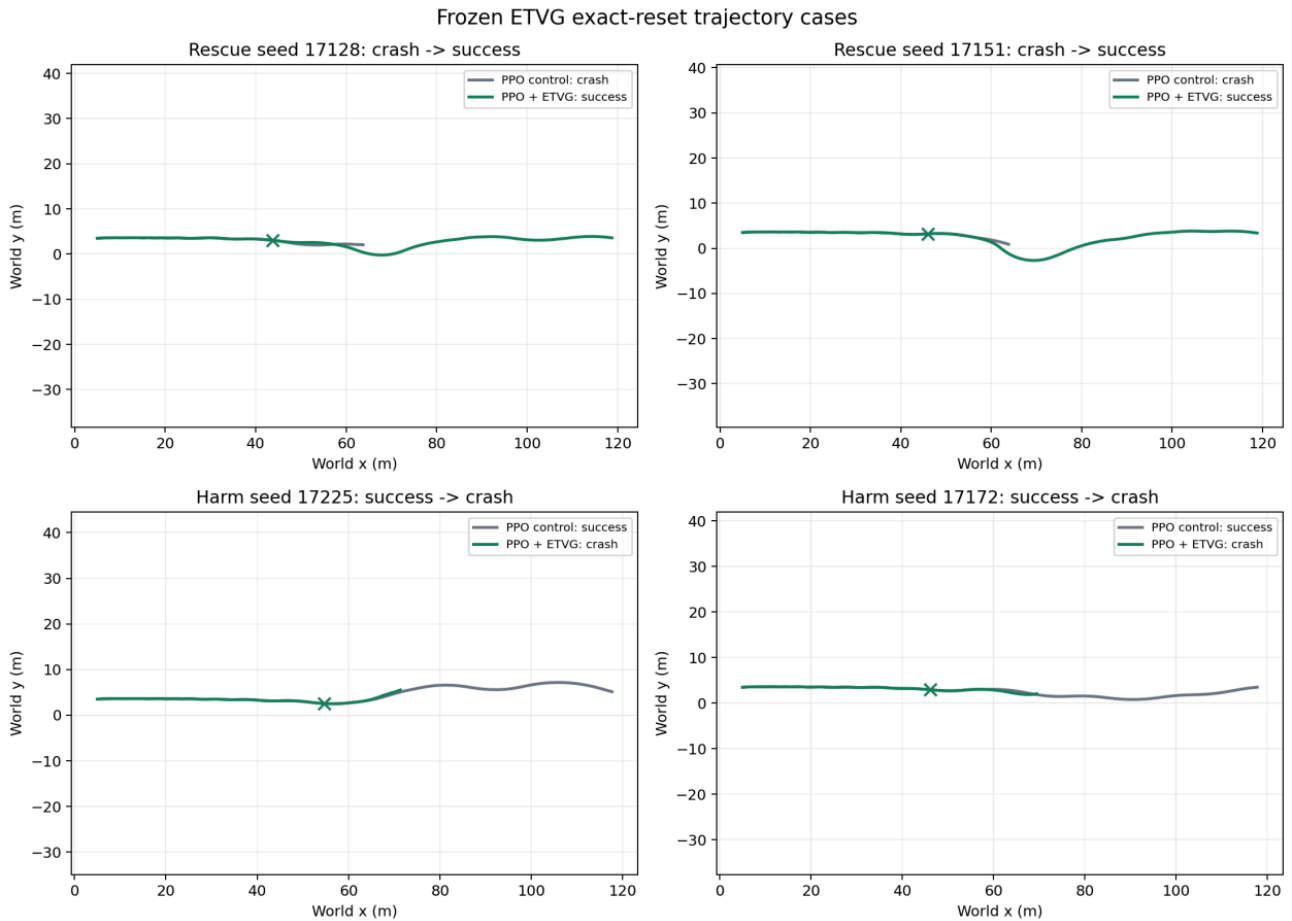


Figure 3. Two rescues and two harms under the frozen ETVG rule; paired paths begin from the same simulator state, and crosses indicate the first accepted event

Exact-reset trajectory cases provide a qualitative validation of the same mechanism. Candidate rescues are ranked by paired route-completion gain and harms by lower event probability; one

nonreproducible replay is excluded. In the two crash-to-success rescues and two success-to-crash harms, the rule overrides five longitudinal actions, preserves steering, processes one terminal frame, and has no render error. Figure 3 illustrates the aggregate result; it does not re-estimate performance from selected examples.

5.4. Frozen-rule traffic-density shift

The frozen rule retains its benefit under heavier traffic at the same intersection. At density 0.8, Control/ETVG success is 0.467/0.547, with 27 rescues and 3 harms ($p=8.43e-6$); collision falls from 0.517 to 0.447, out-of-road from 0.017 to 0.007, and route completion rises from 0.755 to 0.785. The event rate is 0.493, with 16.91 process-frame inferences and 2.38 overridden steps per episode. Seed, terminal, rendering, and steering audits pass. This same-intersection density shift does not establish new-geometry generalization.

Local Qwen2.5-VL-7B is a negative ablation. Its original four-score prompt returns one vector for all 24 first-event frames. On 42 action-sensitive forks, it always selects unchanged control when state summaries are included; without numerical cues, a 10-frame calibration probe becomes constant cap3. Mean latency is 12.5 s for semantic verification and 13.4 s for four-action selection, versus 33.7 ms per image for ResNet18. Frequent local-VLM control is thus slow and uninformative here. This result applies only to the tested local 7B model and prompts, not to larger VLMs generally.

After a 100k MaskablePPO stage, matched control, mask-disabled policy, and the complete visual system reach 0.330/0.347/0.440 success. Most of the gain disappears without the mask, indicating that the visual behavior is not fully distilled into PPO. Grad-CAM remains a qualitative, rather than causal, audit [18]: it can show where the detector is sensitive, but it cannot establish that a highlighted region caused the action-level improvement.

Across 20 episodes, simulation runs at 75.30 steps/s without vision and 39.33 when processing one frame every five steps. ResNet18 occupies 42.8 MiB and averages 33.7 ms per image; interval 5 uses 16.4 calls per episode versus 82.5 for every-step processing.

6. Discussion

6.1. Interpretation of the results

The experiments distinguish reward shaping from action guidance. State terms alter behavior but do not yield monotonic improvements in success. Terminal Safety trades fewer road departures for more collisions; Comfort improves its speed-change proxy but reduces route completion; the two multi-objective settings also perform worse than Baseline. The dense scalar visual reward is the clearest alignment failure because it achieves the highest shaped return while completing the fewest routes in its matched comparison.

ETVG instead binds one visual event to one restricted action. Exact-reset branching produces more rescues than harms, RGB approaches the privileged State Trigger under the same cap, and the effect appears in all three PPO lineages and at a density of 0.8. Thresholds of 0.50-0.80 behave similarly, and every-step processing does not outperform five-step intervals. Together, these controls support an event-action interpretation without claiming that visual information is universally beneficial.

6.2. Scope of the hybrid controller

ETVG is intentionally hybrid. State PPO handles route following and steering, while RGB contributes a single interaction event and cannot choose a route or lateral action. This is not end-to-end visual driving; it is that a current-image event can improve a matched policy through a narrow, transparent intervention.

The terminal image is processed once to verify outcome coverage but cannot alter preceding actions. The improvement therefore comes from process-frame events, not from classifying an outcome after it occurs.

6.3. Relationship between ResNet and the VLM

The local VLM is dominated by prompt priors on action-sensitive samples and is too slow for frequent queries. ResNet18 is less general, but it targets one measurable event, runs in milliseconds, and enables episode-disjoint testing. The result favors a specialized online signal for this task rather than a general rejection of VLM supervision. A larger or better calibrated VLM could still contribute offline labels, explanations, or low-frequency review, if its contribution is verified independently of prompt fluency.

6.4. Limitations

All performance evidence comes from MetaDrive. Overlay cues assume a perception/planning stack, and overlay-trained weights do not transfer to raw renderings. Offline labels depend on simulator geometry and arrival-time estimates, so oracle errors propagate to detector targets. The cap has no formal safety guarantee and causes 29 harms as well as 94 rescues. The density test changes traffic, not geometry; no valid new-geometry test is available.

The state-reward comparison uses one PPO seed and two fixed MultiReward settings; ETVG uses three seeds. Three seeds improve repeatability over one checkpoint but remain a limited sample of training randomness. Broader claims require independent annotations, raw-camera training, competent source policies on additional geometries, another simulator, and real-world perception noise. The artifact records the frozen detector and rule, seed ranges, configurations, canonical results, runtime audits, and claim boundaries; the supported controller requires no commercial API.

7. Conclusion

Matched 1.8-million-step experiments show that a dense scalar visual reward can raise return while reducing success from 0.388 to 0.287. ETVG replaces that score with one RGB-derived yielding event and one short longitudinal intervention. Across three PPO seeds, mean success rises from 0.338 to 0.410 while collision and road-departure rates fall; the frozen density-0.8 test raises success from 0.467 to 0.547. The evidence supports bounded event-triggered visual assistance for this intersection task. Raw-image transfer, new geometries, independent labels, more training seeds, and another simulator remain future work.

References

- [1] Li, Q., Peng, Z., Feng, L., Zhang, Q., Xue, Z., & Zhou, B. (2023). MetaDrive: Composing diverse driving scenarios for generalizable reinforcement learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3), 3461–3475. <https://doi.org/10.1109/TPAMI.2022.3190471>

- [2] Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017). CARLA: An open urban driving simulator. *Proceedings of the 1st Annual Conference on Robot Learning*, PMLR 78, 1–16.
- [3] Shah, S., Dey, D., Lovett, C., & Kapoor, A. (2018). AirSim: High-fidelity visual and physical simulation for autonomous vehicles. In Hutter, M., & Siegwart, R. (Eds.), *Field and Service Robotics*, 621–635. https://doi.org/10.1007/978-3-319-67361-5_40
- [4] Kiran, B. R., Sobh, I., Talpaert, V., et al. (2022). Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6), 4909–4926. <https://doi.org/10.1109/TITS.2021.3054625>
- [5] Codevilla, F., Muller, M., Lopez, A., Koltun, V., & Dosovitskiy, A. (2018). End-to-end driving via conditional imitation learning. *Proceedings of the IEEE International Conference on Robotics and Automation*, 4693–4700. <https://doi.org/10.1109/ICRA.2018.8460487>
- [6] Chen, Y., Chen, M., Zhang, Q., et al. (2026). ROAD: Responsibility-oriented reward design for reinforcement learning in autonomous driving. *IEEE Robotics and Automation Letters*, 11(6), 6552–6559. <https://doi.org/10.1109/LRA.2026.3678458>
- [7] Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv:1707.06347.
- [8] Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *Proceedings of the International Conference on Machine Learning*, 1861–1870.
- [9] Ng, A. Y., Harada, D., & Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. *Proceedings of the International Conference on Machine Learning*, 278–287.
- [10] Baumli, K., Baveja, S., Behbahani, F. M. P., et al. (2023). Vision-language models as a source of rewards. arXiv:2312.09187.
- [11] Wang, Y., Sun, Z., Zhang, J., et al. (2024). RL-VLM-F: Reinforcement learning from vision language foundation model feedback. *Proceedings of the International Conference on Machine Learning*, PMLR 235, 51484–51501.
- [12] Bai, S., Chen, K., Liu, X., et al. (2025). Qwen2.5-VL technical report. arXiv:2502.13923.
- [13] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 770–778. <https://doi.org/10.1109/CVPR.2016.90>
- [14] Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). ImageNet: A large-scale hierarchical image database. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 248–255. <https://doi.org/10.1109/CVPR.2009.5206848>
- [15] Alshiekh, M., Bloem, R., Ehlers, R., Könighofer, B., Niekum, S., & Topcu, U. (2018). Safe reinforcement learning via shielding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1), 2669–2678. <https://doi.org/10.1609/aaai.v32i1.11797>
- [16] Huang, S., & Ontañón, S. (2020). A closer look at invalid action masking in policy gradient algorithms. arXiv:2006.14171.
- [17] Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-Baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268), 1–8.
- [18] Selvaraju, R. R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., & Batra, D. (2017). Grad-CAM: Visual explanations from deep networks via gradient-based localization. *Proceedings of the IEEE International Conference on Computer Vision*, 618–626. <https://doi.org/10.1109/ICCV.2017.74>