EgoEvolve: A Novel Perception-Control Coordination for IoV-Enabled Autonomous Driving

Chuanyu Ju
Fudan University
China
24210860049@
m.fudan.edu.cn

Lang Qian
Fudan University
China
21110860013@
m.fudan.edu.cn

Jingyu Zhang
Fudan University
China
jingyuzhang22@
m.fudan.edu.cn

Jiayue Jin
Fudan University
China
24110860008@
m.fudan.edu.cn

Azzedine Boukerche
University of Ottawa
Canada
aboukerche@gmail.com

Peng Sun
Duke Kunshan University
China
peng.sun@ieee.org

Liang Song
Fudan University
China
songl@fudan.edu.cn

***Abstract*—Reinforcement learning (RL) trains autonomous-driving policies through repeated interaction with the environment. In visual RL, however, the perceptual encoder is often trained mainly from delayed scalar rewards. These rewards provide little evidence about which representation errors later contribute to driving failure. The issue is more pronounced in connected driving, where the policy must combine external information with responses to changing traffic. We introduce EgoEvolve, a visual RL approach that leverages Online Evolutive Learning (OEL) to turn the agent's own closed-loop outcomes into auxiliary supervision for the shared visual encoder. Given the current observation-action pair, EgoEvolve predicts short-horizon ego dynamics, route progress, reward-related feedback, safety risk, and action quality. The Proximal Policy Optimization (PPO) control backbone and the Vision-Language Model (VLM)-based reward remain unchanged. Experiments in CARLA indicate that EgoEvolve improves both training stability and fixed-route driving performance over a strong VLM-RL baseline. When checkpoints are selected under a safety-prioritized rule, the method attains a 6.63% Relative Score gain with full route completion, zero collisions, and a 1.00 success rate. The OEL branch is removed at inference time, so the deployed policy stays lightweight.**

***Keywords*—Autonomous driving; reinforcement learning; perception-control coordination; EgoEvolve; online evolutive learning; Internet of Vehicles; low-latency decision making; collaborative perception.**

I. Introduction

The Internet of Vehicles (IoV) extends autonomous driving perception beyond onboard sensors. Vehicles can exchange information with other vehicles and roadside infrastructure, helping the ego vehicle observe occluded or distant traffic. Representative methods such as OPV2V, V2X-ViT, Where2comm, V2VNet, DiscoNet, and CoBEVT have explored cooperative feature fusion, communication-efficient perception, and cross-agent representation learning [1]–[6].

In connected driving, however, more information does not automatically lead to better control. The ego vehicle must select information according to the driving task, which tightly couples perception, information selection, and control [7]. A representation that performs well on perception accuracy may still fall short for closed-loop driving, because it does not reveal how the current scene and action will affect later outcomes [8].

Visual RL frames driving policy learning as closed-loop optimization: the policy maps visual observations and ego state to continuous control, and CARLA offers a reproducible testbed for this setting [9]–[12]. Earlier driving studies have also looked into conditional imitation, rapid visual policy learning, and implicit affordances [13]–[15]. Scalar rewards, though, give little indication of whether a later failure stems from visual errors, action selection, or the interaction between the two. Reward shaping and reward-function design can supply denser signals, but their terms often hinge on the specific task and on expert choices [16], [17]. Auxiliary objectives, curiosity, contrastive learning, and world models add representation or prediction targets [18]–[23], yet they rarely capture route progress, ego-state changes, reward feedback, and safety risk together.

Vision-language reward methods have used CLIP features or language feedback to construct RL rewards [24]–[28]. VLM-RL applies this idea to safe autonomous driving [29]. Related driving studies have examined safety improvement, latent reward design, parameterized skills, scenario generation, and reward evolution [30]–[34]. EgoEvolve takes a different route: it keeps the VLM-based reward unchanged and adds outcome-conditioned supervision to the shared visual encoder.

We introduce EgoEvolve, a visual RL approach built on Online Evolutive Learning (OEL) for outcome-conditioned perception supervision. OEL has been studied as an algorithmic-level technology for online extraction of high-quality data and evolutive adaptation of heterogeneous models toward application-specific intelligence [7], [35]–[37]. In EgoEvolve, online rollouts are used to construct action-conditioned targets for short-horizon route progress, ego-state dynamics, reward feedback, and safety risk. The PPO control backbone and VLM-based reward remain unchanged; the OEL branch only

979-8-3195-0167-7/26/$31.00 ©2026 IEEE

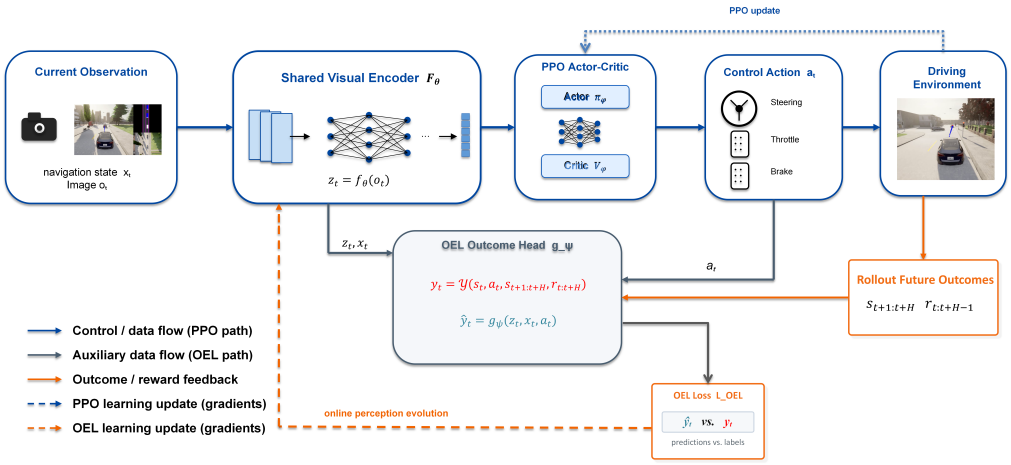


Fig. 1. Overview of the EgoEvolve method: PPO performs closed-loop control, while the OEL branch converts future driving outcomes into training-time perception supervision.

adds supervision to the shared visual encoder.

Our contributions are threefold:

- EgoEvolve uses Online Evolutive Learning to convert closed-loop driving outcomes into online supervision for perceptual representation learning.
- We formulate an action-conditioned future-outcome prediction objective for the shared visual representation. The objective covers route progress, ego-state dynamics, reward variation, and safety risk.
- We validate EgoEvolve in autonomous driving simulation. In the fixed-route CARLA Town02 setting, training-process analysis and ablation studies show improved policy stability and driving efficiency under safety constraints, with a 6.63% gain in Relative Score over a strong VLM-RL baseline.

The overall training and inference paths are shown in Fig. 1.

II. Preliminaries and Problem Formulation

At time t, the agent receives a visual observation o_t and ego-navigation state x_t, and outputs a continuous action a_t for steering, throttle, and braking. The visual encoder produces $z_t = f_\theta(o_t)$, which is then combined with x_t by the PPO actor and critic. In an IoV setting, o_t may come from a fused observation of nearby vehicles or roadside units, so the method can accommodate different collaborative-perception front ends.

The scalar reward evaluates the executed behavior, but it says little about how the current perception-action condition affects later driving. To fill this gap, we define the future rollout after (o_t, x_t, a_t) as

$$\tau_{t:t+H} = \{(s_{t+k}, r_{t+k-1})\}_{k=1}^{H}, \tag{1}$$

where H denotes the prediction horizon, and s_{t+k} contains the future driving state used to construct outcome labels.

From this window, EgoEvolve constructs action-conditioned outcome labels y_t and predicts them from the current representation, ego state, and action:

$$y_t = \{y_{t,j}\}_{j=1}^{J}, \tag{2}$$

$$\hat{y}_t = g_\eta(z_t, x_t, a_t). \tag{3}$$

Here, each component describes route progress, ego-state dynamics, reward feedback, or safety risk. The corresponding prediction loss is $\mathcal{L}_{\text{Ego},t} = \sum_j w_j \ell_j(\hat{y}_{t,j}, y_{t,j})$. This loss supplies separate supervision for collision risk, heading deviation, route progress, and reward, and propagates these signals back to the shared visual encoder.

III. EgoEvolve with Online Evolutive Learning

A. Method Overview

EgoEvolve augments visual PPO with an OEL outcome-prediction branch during training. During rollout collection, the agent records observations, ego states, actions, rewards, and safety events. The OEL head predicts future ego-state dynamics, route progress, reward feedback, safety risk, and action quality, and its loss updates the shared visual encoder. Fig. 1 shows the training path. For IoV-enabled driving, o_t can be a fused Bird's-Eye View (BEV) image, feature map, or semantic observation from a collaborative-perception module.

B. Action-Conditioned Future-Outcome Construction

For each transition, the supervision labels come from future states and rewards after executing a_t, not from the network representation. We set the prediction horizon to $H = 10$. If a sample's 10-step future window crosses an episode boundary or contains a route-termination event, it is discarded. This keeps incomplete rollouts from generating noisy auxiliary labels.

The automatic label construction function is defined as:

$$y_t = \mathcal{Y}(s_t, a_t, s_{t+1:t+H}, r_{t:t+H-1}), \tag{4}$$

TABLE I
CORE NOTATION, LABEL DEFINITIONS, AND OPTIMIZATION LOSSES IN EGOEVOLVE.

Symbol	Outcome label	Loss
Δv_t	$(v_{t+H} - v_t)/s_v$	Smooth L1
Δe_t	$(e_{t+H} - e_t)/s_e$	Smooth L1
$\Delta \psi_t$	$(\psi_{t+H} - \psi_t)/s_\psi$	Smooth L1
Δp_t	$(p_{t+H} - p_t)/s_p$	Smooth L1
R_t^H	$\sum_{k=0}^{H-1} r_{t+k}/s_R$	Smooth L1
c_t	$\mathbb{I}[\text{collision}_{t:t+H}]$	BCE
b_t	$\mathbb{I}[\text{offroad}_{t:t+H}]$	BCE
q_t	bad / neutral / good	CE

where $\mathscr{Y}(\cdot)$ maps the current state-action pair and its future rollout window to a structured outcome label. The label vector is given by:

$$y_t = [\Delta v_t, \Delta e_t, \Delta \psi_t, \Delta p_t, R_t^H, c_t, b_t, q_t], \tag{5}$$

where Δv_t, Δe_t, $\Delta \psi_t$, and Δp_t denote future changes in speed, signed lane offset, signed heading error, and route progress, respectively. R_t^H denotes the short-horizon cumulative reward. The binary variables c_t and b_t indicate collision risk and off-road or trajectory-deviation risk within the future window, while q_t denotes the discretized action-quality category.

Table I summarizes the outcome labels and losses. Continuous labels are normalized and clipped before training. Risk labels are obtained from CARLA termination events and safety states and supervised with Binary Cross-Entropy (BCE). Action-quality labels are discretized from future reward, route progress, lane keeping, and heading stability and supervised with Cross-Entropy (CE).

Taken together, these labels break closed-loop consequences into separate supervision for safety, route alignment, speed change, and reward feedback.

C. Reward-Aware Evolutive Objective

The EgoEvolve loss combines regression terms for continuous outcomes and classification terms for safety and action quality:

$$\mathscr{L}_{\text{Ego}} = w_d \mathscr{L}_{\text{dyn}} + w_r \mathscr{L}_{\text{rew}} + w_s \mathscr{L}_{\text{safe}} + w_q \mathscr{L}_{\text{quality}}. \tag{6}$$

Here, $\mathscr{L}_{\text{dyn}}$, $\mathscr{L}_{\text{rew}}$, $\mathscr{L}_{\text{safe}}$, and $\mathscr{L}_{\text{quality}}$ measure errors for dynamics and route progress, reward, safety, and action quality, respectively.

The overall training objective is defined as:

$$\mathscr{L} = \mathscr{L}_{\text{PPO}} + \lambda_{\text{Ego}} \mathscr{L}_{\text{Ego}}. \tag{7}$$

where λ_{Ego} controls the auxiliary weight. Auxiliary gradients update f_θ and g_η, while the actor-critic heads are optimized by PPO. In this way, the auxiliary task connects the current visual-action condition to future reward, safety, route progress, and ego-state changes.

At inference, g_η is removed and only the PPO policy is used. The auxiliary branch therefore improves training without adding to the computational cost of deployment-time control.

TABLE II
TRAINING-PROCESS EFFICIENCY COMPARISON.

Method	Stable Step↓	Average Speed↑			Safety Cost↓		
		0–300k	300–600k	600k–1M	CR	DCF	TCF
VLM-RL	–	15.55	16.80	17.11	0.93	5.07	1.52
EgoEvolve	48.1k	17.51	17.85	17.68	0.42	1.40	0.59

Stable Step is the first step at which 50 consecutive logs satisfy average speed $\geq$ 17.0 and collision rate $\leq$ 0.50. CR, DCF, and TCF denote collision rate, distance-to-centerline failure, and trajectory-control failure, respectively. The safety costs are computed over the 900k–1M stage.

TABLE III
TOWN02 FIXED-ROUTE TESTING COMPARISON.

Model	AS↑	RC↑	TD↑	CS↓	SR↑
VLM-RL [29]	19.30	0.97	2028.2	0.02	0.93
EgoEvolve	20.82	1.00	2103.51	0.00	1.00

IV. EXPERIMENTS AND RESULTS

A. Experimental Setup

We compare EgoEvolve with VLM-RL-PPO. The PPO control backbone and VLM-based semantic reward are unchanged; only the OEL branch is added during training. Experiments are conducted in the CARLA visual driving setting with local or fused BEV observations. We report average speed (AS), route completion (RC), total distance (TD), collision score (CS), success rate (SR), and Relative Score (RS). Among these, AS, RC, TD, and SR should be as high as possible, while CS reflects safety and should stay low. RS is computed after safety-prioritized checkpoint selection:

$$\text{RS} = 0.7 \cdot \frac{\text{AS}}{\text{AS}_{\text{ref}}} + 0.3 \cdot \frac{\text{TD}}{\text{TD}_{\text{ref}}}. \tag{8}$$

The reference values are $\text{AS}_{\text{ref}} = 19.3$ and $\text{TD}_{\text{ref}} = 2028.2$. Checkpoints are first screened by RC, CS, and SR; the checkpoint with the highest AS among the best safety and completion profiles is selected.

B. Main Results

In the training logs, EgoEvolve reaches the defined speed-safety criterion at 48.1k steps, while VLM-RL does not meet it under the same threshold. EgoEvolve also has higher average speed in the three reported training stages and lower safety cost in the 900k–1M stage.

Table II shows that EgoEvolve reaches the defined criterion earlier and has higher average speed in all three training stages. Its three reported safety costs are also lower in the final training stage.

Table III shows that EgoEvolve improves AS from 19.30 to 20.82 and RS from 1.0000 to 1.0663, a 6.63% gain. It also keeps full route completion, zero collision score, and a 1.00 success rate under the safety-prioritized rule.

Table IV summarizes the reported cross-town results. EgoEvolve has higher AS than VLM-RL in Town01, Town02, and Town04. In Town03 and Town05, its AS is lower, but route completion and safety remain better in the reported tests. The main improvement across towns is therefore completion and safety stability rather than a uniform speed increase.

TABLE IV
CROSS-TOWN COMPARISON BETWEEN VLM-RL AND EGOEVOLVE.

Town	Model	AS↑	RC↑	TD↑	CS↓	SR↑
Town01	VLM-RL	22.90 ± 0.63	1.00 ± 0.00	5697.6 ± 0.79	0.03 ± 0.05	1.00 ± 0.00
	EgoEvolve	23.57 ± 0.43	1.00 ± 0.00	5686.2 ± 13.2	0.00 ± 0.00	1.00 ± 0.00
Town02	VLM-RL	19.30 ± 1.29	0.97 ± 0.03	2028.2 ± 96.6	0.02 ± 0.03	0.93 ± 0.04
	EgoEvolve	20.82 ± 1.35	1.00 ± 0.00	2103.5 ± 72.7	0.00 ± 0.00	1.00 ± 0.00
Town03	VLM-RL	21.70 ± 0.55	0.91 ± 0.07	3757.8 ± 248.0	1.14 ± 1.54	0.87 ± 0.09
	EgoEvolve	20.55 ± 0.21	0.97 ± 0.00	3924.2 ± 173.0	0.22 ± 0.52	0.93 ± 0.05
Town04	VLM-RL	22.00 ± 3.69	0.80 ± 0.17	12684.1 ± 3608.5	2.15 ± 1.59	0.70 ± 0.28
	EgoEvolve	23.67 ± 0.18	0.97 ± 0.05	15944.2 ± 1408.7	1.31 ± 1.26	0.97 ± 0.06
Town05	VLM-RL	22.90 ± 0.86	0.93 ± 0.03	3322.5 ± 203.4	0.46 ± 0.54	0.87 ± 0.05
	EgoEvolve	22.04 ± 0.35	1.00 ± 0.00	3733.1 ± 212.6	0.00 ± 0.00	1.00 ± 0.00

C. IoV Deployment Considerations

The current experiments use a single-agent CARLA setting with local or fused BEV observations; they do not evaluate explicit V2X message exchange. In an IoV deployment, upstream collaborative perception can replace o_t with fused observations from nearby vehicles or roadside units, while OEL labels remain generated from the ego vehicle's future outcomes. Because the OEL head is removed at inference, the control path does not add auxiliary prediction latency. Communication latency, stale information, and packet loss remain to be evaluated in future V2X experiments.

D. Ablation Study

The ablation study removes or changes one supervision source at a time. No-reward OEL tests future-outcome prediction without reward labels, Interaction OEL adds local interaction labels, Actor-conditioned OEL feeds auxiliary outputs to the actor, and Quality-guided OEL adds action-quality classification. Table V shows that the full combination gives the highest RS; adding interaction labels alone gives only a small gain.

TABLE V
DIAGNOSTIC ABLATION RESULTS OF EGOEVOLVE.

Method	Step	AS	RC	CS	SR	RS
VLM-RL	–	19.30	0.97	0.02	0.93	1.0000
No-reward OEL	1000k	19.74	1.00	0.00	1.00	1.0272
Interaction OEL	500k	19.10	1.00	0.00	1.00	1.0039
Actor-conditioned OEL	500k	20.50	1.00	0.00	1.00	1.0554
Quality-guided OEL	800k	20.30	1.00	0.00	1.00	1.0473
EgoEvolve	800k	20.82	1.00	0.00	1.00	1.0663

Step denotes the best checkpoint selected for each method under the safety-prioritized checkpoint selection rule.

V. CONCLUSION

This paper presented EgoEvolve, a visual RL method that adopts OEL to supervise perception with short-horizon driving outcomes in IoV-enabled autonomous driving. In CARLA, it improves speed and Relative Score over VLM-RL while maintaining route completion and safety under the selection rule; the ablation results favor combining future ego outcomes, route progress, safety events, and reward supervision. The OEL branch is removed at inference, leaving the deployment-time control path unchanged; future work will test explicit V2X perception and communication-aware driving.

ACKNOWLEDGMENTS

This work was supported by National Key Research and Development Program of China, Project No. 2024YFE0200700, Subject No. 2024YFE0200703. This work was also supported in part by the Specific Research Fund of the Innovation Platform for Academicians of Hainan Province under Grant YSPTZX202314, and in part by the Shanghai Key Research Laboratory of NSAI.

REFERENCES

[1] R. Xu et al., "OPV2V: An open benchmark dataset and fusion pipeline for perception with vehicle-to-vehicle communication," in Proc. IEEE Int. Conf. Robotics and Automation, 2022, pp. 2583–2589.

[2] R. Xu et al., "V2X-ViT: Vehicle-to-everything cooperative perception with vision transformer," in Proc. Eur. Conf. Computer Vision, 2022, pp. 107–124.

[3] Y. Hu et al., "Where2comm: Communication-efficient collaborative perception via spatial confidence maps," in Proc. Adv. Neural Inf. Process. Syst., 2022.

[4] T.-H. Wang et al., "V2VNet: Vehicle-to-vehicle communication for joint perception and prediction," in Proc. Eur. Conf. Computer Vision, 2020, pp. 605–621.

[5] Y. Li et al., "Learning distilled collaboration graph for multi-agent perception," in Proc. Adv. Neural Inf. Process. Syst., 2021.

[6] R. Xu et al., "CoBEVT: Cooperative bird's eye view semantic segmentation with sparse transformers," in Proc. Conf. Robot Learning, 2022.

[7] J. Jin, L. Qian, and L. Song, "Multi-agent framework based on coordinated sensing and control with online evolutive learning," in Proc. IEEE 10th World Forum Internet Things (WF-IoT), 2024.

[8] J. Yamada, K. Pertsch, A. Gunjal, and J. J. Lim, "Task-induced representation learning," in Proc. Int. Conf. Learning Representations, 2022.

[9] B. R. Kiran et al., "Deep reinforcement learning for autonomous driving: A survey," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 6, pp. 4909–4926, 2022.

[10] A. Haydari and Y. Yilmaz, "Deep reinforcement learning for intelligent transportation systems: A survey," IEEE Trans. Intell. Transp. Syst., vol. 23, no. 1, pp. 11–32, 2022.

[11] J. Schulman et al., "Proximal policy optimization algorithms," arXiv:1707.06347, 2017.

[12] A. Dosovitskiy et al., "CARLA: An open urban driving simulator," in Proc. Conf. Robot Learning, 2017, pp. 1–16.

[13] F. Codevilla et al., "End-to-end driving via conditional imitation learning," in Proc. IEEE Int. Conf. Robotics and Automation, 2018, pp. 4693–4700.

[14] A. Kendall et al., "Learning to drive in a day," in Proc. IEEE Int. Conf. Robotics and Automation, 2019, pp. 8248–8254.

[15] M. Toromanoff, E. Wirbel, and F. Moutarde, "End-to-end model-free reinforcement learning for urban driving using implicit affordances," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2020, pp. 7153–7162.

[16] A. Y. Ng et al., "Policy invariance under reward transformations: Theory and application to reward shaping," in Proc. Int. Conf. Machine Learning, 1999, pp. 278–287.

[17] A. Abouelazm, J. Michel, and J. M. Zollner, "A review of reward functions for reinforcement learning in the context of autonomous driving," arXiv:2405.01440, 2024.

[18] M. Jaderberg et al., "Reinforcement learning with unsupervised auxiliary tasks," in Proc. Int. Conf. Learning Representations, 2017.
[19] D. Pathak et al., "Curiosity-driven exploration by self-supervised prediction," in Proc. Int. Conf. Machine Learning, 2017, pp. 2778–2787.
[20] M. Laskin, A. Srinivas, and P. Abbeel, "CURL: Contrastive unsupervised representations for reinforcement learning," in Proc. Int. Conf. Machine Learning, 2020, pp. 5639–5650.
[21] D. Ha and J. Schmidhuber, "World models," arXiv:1803.10122, 2018.
[22] Y. Li et al., "Enhancing end-to-end autonomous driving with latent world model," arXiv:2406.08481, 2024.
[23] J. Jin et al., "EvoPSF: Online evolution of autonomous driving models via planning-state feedback," arXiv:2508.11453, 2025.
[24] A. Radford et al., "Learning transferable visual models from natural language supervision," in Proc. Int. Conf. Machine Learning, 2021, pp. 8748–8763.
[25] K. Baumli et al., "Vision-language models as a source of rewards," in Second Agent Learning in Open-Endedness Workshop at NeurIPS, 2023.
[26] S. Sontakke et al., "RoboCLIP: One demonstration is enough to learn robot policies," Adv. Neural Inf. Process. Syst., vol. 36, 2024.
[27] J. Rocamonde et al., "Vision-language models are zero-shot reward models for reinforcement learning," in Proc. Int. Conf. Learning Representations, 2024.
[28] X. Ye et al., "LORD: Large models based opposite reward design for autonomous driving," arXiv:2403.18965, 2024.
[29] Z. Huang, Z. Sheng, Y. Qu, J. You, and S. Chen, "VLM-RL: A unified vision language models and reinforcement learning framework for safe autonomous driving," arXiv:2412.15544, 2024.
[30] Z. Cao et al., "Trustworthy safety improvement for autonomous driving using reinforcement learning," Transp. Res. Part C Emerg. Technol., vol. 138, p. 103656, 2022.
[31] J. Chen, S. E. Li, and M. Tomizuka, "Interpretable end-to-end urban autonomous driving with latent deep reinforcement learning," IEEE Trans. Intell. Transp. Syst., vol. 23, 2022.
[32] L. Wang et al., "Efficient reinforcement learning for autonomous driving with parameterized skills and priors," in Proc. Robotics: Science and Systems, 2023.
[33] J. Zhang, C. Xu, and B. Li, "ChatScene: Knowledge-enabled safety-critical scenario generation for autonomous vehicles," in Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit., 2024, pp. 15459–15469.
[34] R. Hazra et al., "Revolve: Reward evolution with large language models for autonomous driving," arXiv:2406.01309, 2024.
[35] L. Song, X. Hu, G. Zhang, P. Spachos, K. Plataniotis, and H. Wu, "Networking systems of AI: On the convergence of computing and communications," IEEE Internet Things J., vol. 9, no. 20, pp. 20352–20381, 2022.
[36] L. Qian, J. Jin, P. Sun, J. Liu, B. Hu, A. Boukerche, and L. Song, "A coordinated optimization framework for intelligent agents with online evolutive learning," IEEE Internet Things J., early access, 2026.
[37] D. Li, X. Zhu, and L. Song, "Mutual match for semi-supervised online evolutive learning," Applied Intelligence, vol. 53, pp. 3336–3350, 2023.