Autonomous Driving Based on Imitation and Active Inference

Sheida Nozari[1,2](✉), Ali Krayani[1], Pablo Marin[2], Lucio Marcenaro[1],
David Martin[2], and Carlo Regazzoni[1]

[1] University of Genoa, Genoa, Italy
sheida.nozari@edu.unige.it
[2] Carlos III University of Madrid, Getafe, Spain

Abstract. We advance a novel computational model of acquiring a hierarchical action sequence and its use for minimizing the divergence between observation and prediction. The model is grounded in a principled framework to learn and cognize the dynamic surrounding of a learning agent, which exemplifies the model's functioning by presenting a simulation of overtaking scenarios in autonomous driving. The learning agent integrates imitation learning and active inference to form hierarchical representations from expert demonstrations. During the online learning phase, the learning agent improves the action selection procedure based on the prior knowledge (exploiting) and the novel interactions with a dynamic environment (exploring). The proposed method applies an active knowledge sampling during the learning agent's movements to make a dynamic inference in the Bayesian structure by message passing through the multi-levels. A line-changing driving scenario with different levels of complexity is organized to verify the proposed framework's efficiency by dealing with single and multiple dynamic objects interacting in the environment.

Keywords: Imitation Learning · Motion prediction · Active inference · Dynamic Bayesian Network

1 Introduction

As autonomous vehicle (AV) technology progresses towards full autonomy, self-driving vehicles are more likely to become a reality in the coming years [1]. Although AVs have demonstrated a guarantee to operate in a structured environment under reasonable driving circumstances, they are still facing unpredictable behaviours causing disastrous consequences in unseen environments with high uncertainty [2]. A crucial component of autonomous driving is making driving decisions based on the vehicle's environmental changes and adapting to different types of driving conditions [3].

Due to their heavy manual tuning in modelling dynamic interactions among vehicles, traditional planning methods are not always scalable and cost-effective

M. Valle et al. (Eds.): SYSINT 2022, LNNS 546, pp. 13–22, 2023.
https://doi.org/10.1007/978-3-031-16281-7_2

[4]. Imitation Learning (IL) offers an alternative solution that mimics an expert's driving behaviour based on a data-driven approach. However, in IL, one of the primary challenges is specifying how an AV acts when it is outside the presented states by the demonstrations [5]. Reinforcement Learning (RL) can overcome such an issue by exploring new states and actions while an agent is interacting with the environment. Exploration plays a vital role in RL, however, it typically requires a large amount of training data and computational resources to succeed [6]. In response to this challenge, IL leverages inverse reinforcement learning (IRL) to deduce the reward function from demonstrations [7]–[8]. Learning a parameterized reward function provides a compact representation of the demonstrator's preferences and allows policy optimization to generalize to the unnoticed states. However, IRL methods always generate uncertainty about the proper reward function, which can have adverse outcomes if the learning agent figures a reward function leading it to learn the wrong policy. Additionally, systems that uses deterministic models rather than probabilistic ones ignore the stochastic property of dynamic environments. We propose an IL approach combined with a hierarchical probabilistic model under the Active Inference framework [9] to overcome the mentioned limitations. An active imitator agent (i.e., an AV) aims to learn a robust policy to uncertainty and dynamic environmental changes while accomplishing a specific task. The agent can effectively trade-off the exploration-exploitation rate with the expected return within active inference. The latter allows an AV to detect novel situations by comparing what the agent is expecting to observe based on the rules learned from an expert's demonstration and what it is actually observing to decide whether to explore new actions or exploit what has been learned by imitating the expert. AV aims to learn a sequence of actions that lead to the minimization of the abnormality. The main contributions of this work are: *1)* Integrate Active inference to IL to optimize the learning policy by establishing the balance between expected return and abnormality value under the new experiences. *2)* Optimize the exploration-exploitation rate by distinguishing the normal and abnormal situations through the Bayesian predictive and diagnostic messages. *3)* Optimize the action planning concerning the uncertainty and minimize the imitation loss with respect to expert demonstrations. *4)* Demonstrate that the proposed framework achieves better results in terms of learning aspects than existing RL methods.

2 Proposed Framework

The proposed framework is divided into two main phases: offline and online learning. During the first phase, we learn a situation model explaining how an *expert agent* (E) and a *dynamic Object* (O) interact in the environment. Moreover, we organize a First-Person generative model (FP), allowing a *learning agent* (L) to learn the E's behavior by observing its demonstrations. In the second phase, L modifies and updates its imperfect knowledge from the sub-optimal expert demonstration through an Active First-Person generative model (AFP) with respect to its relative distance with a moving object ($\hat{\text{O}}$) in a dynamic environ-

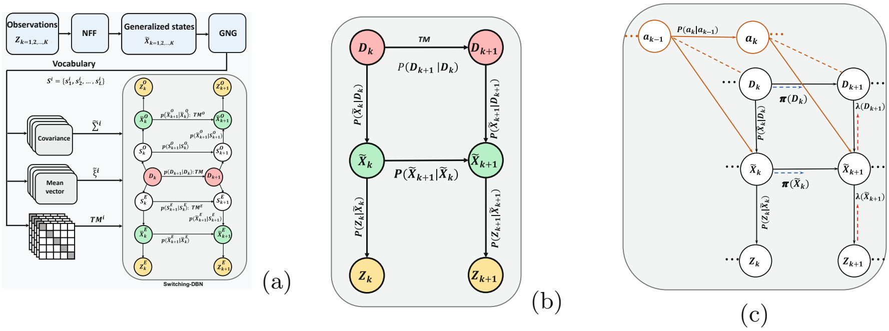

Fig. 1. Learning models: a) Situation model, b) FP model, c) AFP model.

ment. Both FP and AFP models are structured in Dynamic Bayesian Network (DBN) representations [10].

2.1 Offline Learning Phase

Situation Model. This model is structured to explain the interaction among two dynamic agents (see Fig. 1-(a)), E and O by using a switching DBN [11]. In order, the variables $\mathrm{Z}_k^{\mathrm{E}}$ and $\mathrm{Z}_k^{\mathrm{O}}$ illustrate the agents observations at the low level of the hierarchy (yellow nodes). The middle level (green nodes), describes the joint hidden continuous Generalized states (GSs) containing the dynamics of two agents at each time instant k as follows: $\tilde{\mathbf{X}}_k = [\tilde{\mathrm{X}}_k^{\mathrm{E}}\ \tilde{\mathrm{X}}_k^{\mathrm{O}}]^\intercal$, where $\tilde{\mathrm{X}}_k^{\mathrm{E}}$, $\tilde{\mathrm{X}}_k^{\mathrm{O}}$ denote the GSs of E and O, respectively. The GS related to agent i is defined as a vector composed of the agent's state and its first-order temporal derivative, such that $\tilde{\mathrm{X}}_k^i = [\boldsymbol{x}\ \dot{\boldsymbol{x}}]^\intercal$ where $\boldsymbol{x} \in \mathbb{R}^d$, $\dot{\boldsymbol{x}} \in \mathbb{R}^d$, $i \in \{\mathrm{E}, \mathrm{O}\}$ and d stands for the dimensionality. The correlation between Z_k^i and $\tilde{\mathrm{X}}_k^i$, which describes the observation model, is defined as:

$$\mathrm{Z}_k^i = \mathrm{H}\tilde{\mathrm{X}}_k^i + v_k, \tag{1}$$

where $H = [\mathrm{I}_d\ \ 0_{d,d}]$ stands for the observation matrix that measures of how dependent the measurements (Z_k^i) are upon the hidden GSs ($\tilde{\mathrm{X}}_k^i$) and $v_k \sim \mathcal{N}(0, \mathrm{R})$ is the measurement noise that follows a zero-mean Gaussian distribution with covariance R.

Initially, the evolution of $\tilde{\mathbf{X}}_k$ is assumed to follow a static equilibrium assumption described by:

$$\tilde{\mathrm{X}}_k^i = \mathrm{A}\tilde{\mathrm{X}}_{k-1}^i + w_k, \tag{2}$$

where $\mathrm{A} \in \mathbb{R}^{d \times d}$ and $w_k \sim \mathcal{N}(0, \mathrm{Q})$ depict the dynamic matrix and the process noise, respectively. $\tilde{\mathbf{X}}_k$ is predicted by utilizing a Null Force Filter (NFF) according to (2). The NFF calculates the innovations that encode the deviations between predictions and observations as : $\varepsilon_{\tilde{\mathrm{X}}_t} = \mathrm{H}^{-1}\big(\mathrm{Z}_k^i - \mathrm{H}\tilde{\mathrm{X}}_k^i\big)$. The Growing Neural Gas with utility measurement (GNG-U) [12] is empolyed to cluster in an unsupervised manner those innovations $\varepsilon_{\tilde{\mathrm{X}}_t}$ that characterise the generalized errors (GEs) and outputs a vocabulary defined as: $\mathbf{S}^i = \{\mathrm{s}_1^i, \mathrm{s}_2^i, \ldots, \mathrm{s}_{\mathcal{L}_i}^i\}$ consisting of $\mathcal{L}_i$ clusters.

The interaction between E and O at multi-level (i.e., discrete and continuous levels) is described by a joint vocabulary (i.e., $\mathbf{S}^{\mathrm{E}}, \mathbf{S}^{\mathrm{O}}$) expressing discrete regions with quasi-linear models that explain the interactive dynamic evolution of joint states over time. Each discrete cluster $\mathrm{s}^i \in \mathbf{S}^i$ follows a multivariate Gaussian distribution with covariance matrix $\tilde{\Sigma}_{\mathrm{s}_k^i}$ and generalized mean value $\tilde{\boldsymbol{\mu}}^{\mathrm{s}^i} = [\mu_{Pos}^{s^i}\ \mu_V^{s^i}]$, where $\mu_{Pos}^{s^i}$ is the states' mean value on positions and $\mu_V^{s^i}$ is the states' mean value on velocity. Thus, a dictionary can be formed as $\mathbf{D} = \{\mathrm{D}_1, \mathrm{D}_2, \cdots, \mathrm{D}_{\mathrm{M}}\}$ according to the learned configurations, where the interaction configuration ($\mathrm{D}_k = [\mathrm{s}_k^{\mathrm{E}}, \mathrm{s}_k^{\mathrm{O}}]^\intercal$) explains the joint activated clusters.

Consequently, an extra vocabulary encoding the joint configurations is defined as: $\mathbf{D} = \{\mathrm{D}_1, \mathrm{D}_2, \cdots, \mathrm{D}_{\mathrm{M}}\}$ with M configurations, where $\mathrm{D}_k \in \mathbf{D}$ and $\mathrm{D}_k = [\mathrm{s}_k^{\mathrm{E}}, \mathrm{s}_k^{\mathrm{O}}]^\intercal$ depicts an interaction configuration explaining the joint activated clusters occurring simultaneously in the agents' vocabularies (red nodes). Each configuration D_k consists of the average position and average velocity of the two agents according to: $\mathrm{D}_k = \big[(\mu_{Pos}, \mu_V)^{\mathrm{E}}, (\mu_{Pos}, \mu_V)^{\mathrm{O}}\big]$. After learning the joint vocabulary, the dynamic model defined in (2) can be updated as follows:

$$\tilde{\mathrm{X}}_k^i = \mathrm{A}\tilde{\mathrm{X}}_{k-1}^i + \mathrm{B}\mu_V^{s_k^i} + w_k, \tag{3}$$

where $\mathrm{B} \in \mathbb{R}^{d \times d}$ stands for the control model matrix, $\mu_V^{\mathrm{s}_k^i} = [\dot{x}_{s_k^i}, \dot{y}_{s_k^i}]$ is a control vector encoding the agent's velocity (on x and y) associated with s_k^i. The Transition Matrix (TM) can be learned by estimating the transition probabilities $\mathrm{P}(\mathrm{D}_{k+1}|\mathrm{D}_k)$ that encodes the dynamic transitions among the learned configurations at the top level of hierarchy is defined as:

$$\mathrm{TM} = \begin{bmatrix} \mathrm{P}(\mathrm{D}_1|\mathrm{D}_1), & \mathrm{P}(\mathrm{D}_1|\mathrm{D}_2), & \ldots, & \mathrm{P}(\mathrm{D}_1|\mathrm{D}_{\mathrm{M}}) \\ \mathrm{P}(\mathrm{D}_2|\mathrm{D}_1), & \mathrm{P}(\mathrm{D}_2|\mathrm{D}_2), & \ldots, & \mathrm{P}(\mathrm{D}_2|\mathrm{D}_{\mathrm{M}}) \\ \vdots & \vdots & \ddots & \vdots \\ \mathrm{P}(\mathrm{D}_{\mathrm{M}}|\mathrm{D}_1), & \mathrm{P}(\mathrm{D}_{\mathrm{M}}|\mathrm{D}_2), & \ldots, & \mathrm{P}(\mathrm{D}_{\mathrm{M}}|\mathrm{D}_{\mathrm{M}}) \end{bmatrix} \tag{4}$$

where $\sum_m^{\mathrm{M}} \mathrm{P}(\mathrm{D}_p|\mathrm{D}_m) = 1$ such that $p, m \in \mathrm{M}$.

First-Person Model. We provide the FP model by using a switching DBN (situation model) represented by a generative DBN to consider the interaction states with a dynamic object ($\hat{\mathrm{O}}$) under the L's interpretation (see Fig. 1-(b)). The discrete level represents the learned configuration $\mathrm{D}_m \in \mathbf{D}$ (red nodes). The continuous level stands for the generalized relative distance between E and O, which will be updated by L and $\hat{\mathrm{O}}$ through interacting in the environment(green nodes). At FP model initialization, the generalized relative distance can be computed by the difference of the joint GS to consider the agent's interaction at a specific configuration at time k, as follow: $\tilde{\mathrm{X}}_k = \big[\tilde{\mathrm{X}}_k^{\mathrm{E}} - \tilde{\mathrm{X}}_k^{\mathrm{O}}\big] = \big[(\boldsymbol{x}^{\mathrm{O}} - \boldsymbol{x}^{\mathrm{E}})(\dot{\boldsymbol{x}}^{\mathrm{O}} - \dot{\boldsymbol{x}}^{\mathrm{E}})\big]$.

At the bottom level (yellow nodes), the observation can be mapped onto observations (Z_k^i) of both agents as: $\mathrm{Z}_k = \left[\mathrm{Z}_k^{\mathrm{E}} - \mathrm{Z}_k^{\mathrm{O}}\right]$.

2.2 Online Learning Phase

In this section, we aim to enhance IL's efficiency through a probabilistic hierarchical model in a dynamic environment. Therefore, the FP model must compound with the active states to minimise the divergence between the internal predictions and observations while acting in the environment.

Active First-Person Model. The AFP model takes as input a sequence of observed or previously learned configurations ($D_{1:M}$), an action based on the current configuration (a_k), and a random variable (Z_k) which represents the learner observation by its exteroceptive sensor at time instant k. The observation provides a relative distance between L and $\hat{\mathrm{O}}$ that is embedded in the continuous level ($\tilde{X}_k$) of the online learning model. L sustains an internal dynamic representation $\mathrm{P}(\mathrm{Z}, \tilde{\mathrm{X}}, \mathrm{D}, a)$ of the external environment encoded in an AFP model (see Fig. 1-(c)) and purposes to implicitly reduce the mismatch between what it is expecting to receive from the environment and what it is actually perceiving.

Action Selection Probability. At each time instant, L evaluates its current situation. In one condition, L has no prior knowledge about the current interaction with $\hat{\mathrm{O}}$. In the other condition, L has been informed that E was in the same situation while interacting with O (prior belief). Thus, L estimates the proper behavior whether to explore by performing new actions based on the current configuration (in the first condition) or exploit from the learnt configuration (in the second condition). The divergence between the observation and prediction is the main criteria which guides the action selection procedure. L employs Particle Filter (PF) to predict the learnt configuration with the least divergence until time k. Likelihood messages ($\lambda(\tilde{\mathrm{X}}_k)$ and $\lambda(\mathrm{D}_k)$) passing in a backward manner from the bottom level towards higher levels inside the AFP enable measuring the anomaly between the prediction of the propagated particles and the learner's observation, which is computed by cosine similarity ($\cos(\theta)$), as follow:

$$\cos(\theta) = \frac{\tilde{\mathrm{Z}}_k \;.\; \tilde{\mathrm{X}}_{k,n}}{||\tilde{\mathrm{Z}}_k|| \; ||\tilde{\mathrm{X}}_{k,n}||}. \tag{5}$$

The computed abnormality assesses the similarity of the current observation with predictions, and the corresponding configuration to the particle with the highest weight (the least anomaly) is assigned as the activated configuration ($\mathring{\mathrm{D}}$). The particles' weights can be updated by using $\lambda(\mathrm{D}_k)$ which is defined as: $\lambda(\mathrm{D}_k) = \lambda(\tilde{\mathrm{X}}_k)\mathrm{P}(\tilde{\mathrm{X}}_k|\mathrm{D}_k)$, where $\lambda(\tilde{\mathrm{X}}_k) = \mathrm{P}(\mathrm{Z}_k|\tilde{\mathrm{X}}_{k,n})$ is a multivariate Gaussian distribution such that $\lambda(\tilde{\mathrm{X}}_k) \sim \mathcal{N}(\mathrm{Z}_k, v_k)$ and $\lambda(\mathrm{D}_k)$ is a discrete probability distribution. Additionally, L considers two parameters to perform an action. One is the exploration rate (ϵ) where the highest particle weight (α) presents a control input on it as follows: $\epsilon_k = 1 - \alpha_k$. Both ϵ_k and α values are in an interval between 0 and 1, as much as α tends to 1 (higher similarity), ϵ_k goes to

0 (less exploration). L learns to decrease the exploration rate while minimizing the distinction between the prediction and observation. The other parameter is a threshold (ρ) which is obtained by a trial-and-error process. L compares ρ and ϵ_k to select an action at time k as below:

$$a_k \sim \begin{cases} \underset{a_k}{\arg\max}\ \mathrm{Q}(\mathcal{A}, \mathring{D}_k), \text{ if } \epsilon < \rho \ \text{ (exploitation)}, \\ \textit{random} \ \text{ from } \mathcal{A}^+, \text{ if } \epsilon \geq \rho \text{ (exploration)}, \end{cases} \tag{6}$$

where $\mathcal{A} = \{\mathcal{A}^{\mathrm{E}}, \mathcal{A}^+\}$, as $\mathcal{A}^{\mathrm{E}} = \{a_1^{\mathrm{E}}, a_2^{\mathrm{E}}, \ldots, a_{\mathrm{Y}}^{\mathrm{E}}\}$ is a set of actions performed by L and encoded in the situation model that L aims to imitate during exploitation and $\mathcal{A}^+ = \{a_1, a_2, \ldots, a_8\}$ is a set of actions realizing 8 cardinal and ordinal directions which L selects during exploration. Moreover, during exploration, L records the new experiencing pair $(\mathrm{D}_k^+, a_k^+ \in \mathcal{A}^+)$ in the Q-table incrementally.

Free Energy Measurement. The AFP model by leveraging a hierarchical structure computes the imitation cost using bottom-up (λ) and lateral messages (predictions by inter-slice links π) that drive posterior agent movements toward a better prediction to optimize the Free Energy (FE), which will be updated after gathering novel observations. During the online phase, L learns an optimized mapping between the Bayesian message passing, which causes a sequence of observations with a minimum distinction between the expectation and likelihood. The distinction can be estimated between the predictive message $\pi(\tilde{\mathrm{X}}_k)$ and the diagnostic message $\lambda(\tilde{\mathrm{X}}_k)$ after performing an action (a_{k-1}). We employ Kullback Leibler-Divergence ($\mathcal{D}_{\mathcal{KL}}$) between $\pi(\tilde{\mathrm{X}}_k)$ and $\lambda(\tilde{\mathrm{X}}_k)$ to calculate the FE after each performed action, as:

$$\mathcal{F} = \mathcal{D}_{\mathcal{KL}}\Big(\lambda(\tilde{\mathrm{X}}_k)||\pi(\tilde{\mathrm{X}}_k)\Big) = \int \lambda(\tilde{\mathrm{X}}_k) \log\left(\frac{\lambda(\tilde{\mathrm{X}}_k)}{\pi(\tilde{\mathrm{X}}_k)}\right) \mathrm{d}\tilde{\mathrm{X}}_k. \tag{7}$$

Action Selection Probability Update. At each time instant k, the model evaluates the performed action (a_{k-1}) with respect to the FE measurement and updates the selection probabilities in the probabilistic Q-table according the following equation:

$$\mathrm{Q}_k^* = (1-\eta)\mathrm{P}(a_{k-1}|\mathrm{D}_{k-1}) + \eta\Big[(1-\mathcal{F}_k) + \gamma \underset{a_k}{\max} \mathrm{P}(a_k|D_k)\Big], \tag{8}$$

where η is the learning rate which controls to what extent the new experiences overrides the previously recorded situations $(1 - \mathcal{F}_k)$ is the normalized reward measurement with a range in $[0, 1]$, and γ is a discount factor. Our objective is to minimize the long term loss by keeping down the $\mathcal{F}$ though improving the action selection procedure.

3 Experimental Evaluation

Our framework is validated using two autonomous vehicles' multisensory information, 'iCab 1' and 'iCab 2' [13]. To consider the lane-changing scenario, the odometry module obtains positional information and velocity from the vehicles, where iCab 2 overtakes iCab 1 from the left side without colliding.

3.1 Offline Learning Phase

This section shows how the situation model is structured by employing NFF as an initial filter on data. The provided GEs by NFF are clustered using GNG that outputs a set of discrete clusters representing the discrete regions of the trajectories generated by E and O. The joint clusters introduce a set of configurations that encode the interactive behavior between the agents. (see Fig. 2).

3.2 Online Learning Phase

The learning agent employs the FP model as a prior beliefs to imitate the expected transitions. During the online phase, by balancing exploration-exploitation trade-off, L's actions are engaged to solve the uncertain aspects of the action selection caused by the new dynamic environment, which adjust the L's hypotheses. The experiments are executed in a simulated environment through 500 episodes with different start positions to train a learning-agent L. Each episode consists of 10 iterations, i.e., L tries 5k iterations by 500 different start positions to learn the policies. We evaluate the performance of the proposed framework and compare it with other learning algorithms from the literature, namely, the general Q-learning, IRL (when an optimal expert is available), and self-learning in the RL context (when optimal expert data is not available).

Performance Evaluation. After trial stage, L acquires knowledge about the contingencies and the likelihood mapping in the generative model is aligned adequate with the reference generative process and the targeted goal (e.g., overtaking the dynamic object). Crucially, we assume that the correctness and accuracy of the action selection procedure guide the learning agent to the expected observations. Figure 3-(a) illustrates that L movements are engaged coherently, which causes less exploration in each trial epoch (e.g., each episode). Additionally, Fig. 3-(a) compares the number of executed actions during the training using different learning methods, where it shows L performs less actions to accomplish its task by using our method than others. Moreover, L adopting the proposed method has higher successful trajectories than other methods as depicted in Fig. 3-(b).

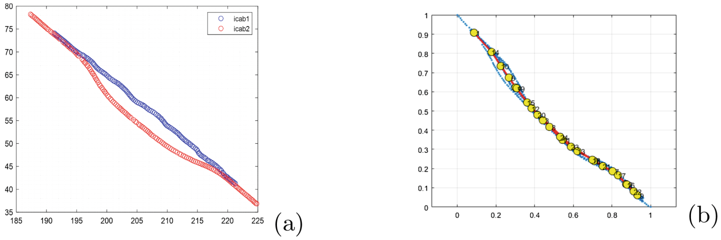

Fig. 2. Learning the situation model. a) iCab2 overtakes iCab1, b) Clustering of GEs.

Another noticeable point is that the AFP model expands its repertoire of action representations by learning new interactions between L and Ô. The novel experienced configurations are recorded in Q-table incrementally, which will be clustered by using GNG after training. Figure 4-(a)-(b) shows the learned model during online phase has an expanded transition matrix than the situation model due to the exploratory aspect of the L's behavior.

Learning Cost Evaluation. The loss cost reduction imperative is one of the components of updating the actions' probabilities (see (8)) that guides the action selection policies in active inference. Figure 5-(a) demonstrates how modifying the actions can reduce the exploration and minimize the imitation cost resulting in a high learning rate during the training phase. Our goal is to find the best set of actions that minimize the imitation loss in terms of FE. Figure 5-(b) shows that the normalized global FE ($\mathcal{F}$) drops down capably to decrease below 0.1. Figure 5-(c) demonstrates that our method outperforms others in terms of suc-

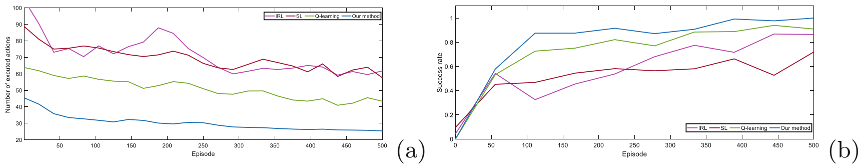

Fig. 3. a) The performed actions by L, b) The success rate.

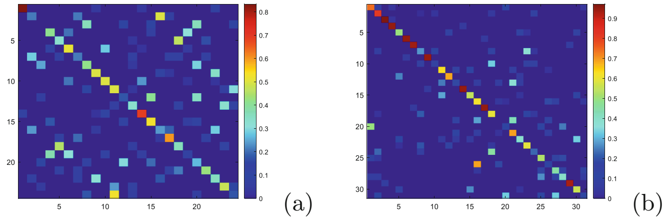

Fig. 4. Incremental learning. a) Corresponding TM to the situation model, b) TM after clustering the learnt Q-table.

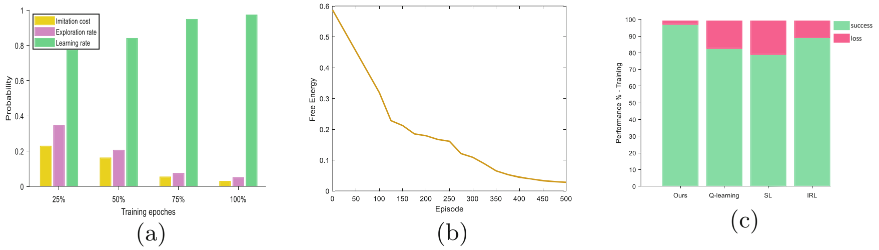

Fig. 5. Learning evaluation. a) The impact of exploration and learning rates after each training quarter on the imitation cost, b) FE measurement, and c) Training result.

cess and loss rate, which is attributed to the effectiveness of motion prediction while dealing with abnormalities that improve the success rate.

During testing, the agent travels through 500 paths with different start positions than the training time. The testing stage includes two levels of difficulties: I) the agent needs to overtake a single dynamic object, and II) the agent needs to overtake multiple dynamic objects. Moreover, each level has three scenarios: overtaking from the left side, overtaking from the right side, and when the agent should decide to overtake from the left or right side of the object(s). Table 1 describes that the adapted agent to the learned model through the presented method can overtake a single object and multiple dynamic objects in the environment effectively, whereas other methods still have a high failure rate. Moreover, the provided vocabulary by E's demonstrations tends to change-line from the left side of O. By experiencing the unseen configuration during the online learning phase, L learns to reduce the collision probability by interacting from the dynamic object's right side. The testing results show that by $5k$ training trajectories, the agent can overcome the experiments with different scenarios where it is necessary i) to overtake from the left side of the dynamic object(s), ii) to overtake from the right side of the dynamic object(s), and iii) to decide to overtake from which side of the dynamic object(s) has less collision probability (mixed situation from both side).

Table 1. Testing the learnt model after $5k$ trial trajectories.

	Our method		Q-learning		IRL		SL	
	Overtake%	Collision%	Overtake%	Collision%	Overtake%	Collision%	Overtake%	Collision%
	Difficulty I: overtaking from one dynamic object							
From the left side	**97.42**	**2.58**	88.13	11.87	90.28	9.72	79.67	20.33
From the right side	**96.81**	**3.19**	84.01	15.99	89.04	10.96	71.52	28.48
From both side	**94.86**	**5.14**	81.35	18.65	86.99	13.01	69.43	30.57
	Difficulty II: overtaking from two dynamic objects							
From the left side	**95.26**	**4.74**	83.78	16.22	89.11	10.89	73.51	26.49
From the right side	**93.13**	**6.87**	80.28	19.72	85.33	14.67	66.17	33.83
From both side	**92.05**	**7.95**	79.15	20.85	80.06	19.94	64.09	35.91

4 Conclusion

A novel framework has been proposed to integrate Imitation Learning with Active Inference for autonomous driving. In the hybrid presented model, the errors between the prediction and observation guide the action selection in two processes. A low amount of error influences L that exploits the prior knowledge to perform an action. In this case, L prepares an imitative response. On the other hand, when experiencing an unobserved configuration causes a high error, L needs to rely on random movements. During the online phase, L learns to decide how to minimize the FE and guide the movements to the imitative actions. Future work concentrates on employing the errors to guide the random movement that might facilitate the execution of congruent actions with the expert demonstrations.

References

1. Claussmann, L., Revilloud, M., Gruyer, D., Glaser, S.: A review of motion planning for highway autonomous driving. IEEE Trans. Intell. Transp. Syst. **21**(5), 1826–1848 (2020)
2. Guo, J., Kurup, U., Shah, M.: Is it safe to drive? an overview of factors, metrics, and datasets for driveability assessment in autonomous driving. IEEE Trans. Intell. Transp. Syst. **21**(8), 3135–3151 (2020)
3. Li, Z.: A hierarchical autonomous driving framework combining reinforcement learning and imitation learning. In: 2021 International Conference on Computer Engineering and Application (ICCEA), pp. 395–400, June 2021
4. Zhou, J.: Exploring imitation learning for autonomous driving with feedback synthesizer and differentiable rasterization. In: 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1450–1457 Sep. 2021
5. Hussein, A., Gaber, M.M., Elyan, E., Jayne, C.: Imitation learning: a survey of learning methods. ACM Comput. Surv. (CSUR) **50**(2), 1–35 (2017)
6. Sutton, R.S. Barto, A.G.: Reinforcement learning: An introduction. MIT press (2018)
7. Ng, A.Y., Russell, S.J., et al.: Algorithms for inverse reinforcement learning. In: Icml, vol. 1, p. 2 (2000)
8. Nozari, S., Marcenaro, L., Martin, D., Regazzoni, C.: Observational learning: imitation through an adaptive probabilistic approach. In: 2021 IEEE International Conference on Autonomous Systems (ICAS IEEE 2021), pp. 1–5
9. Friston, K., FitzGerald, T., Rigoli, F., Schwartenbeck, P., Pezzulo, G.: Active inference: a process theory. Neural Comput. **29**(1), 1–49 (2017)
10. Ghahramani, Z.: Learning dynamic Bayesian networks. In: Giles, C.L., Gori, M. (eds.) NN 1997. LNCS, vol. 1387, pp. 168–197. Springer, Heidelberg (1998). https://doi.org/10.1007/BFb0053999
11. Baydoun, M., Campo, D., Kanapram, D., Marcenaro, L., Regazzoni, C.S.: Prediction of multi-target dynamics using discrete descriptors: an interactive approach. In: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 3342–3346. IEEE (2019)
12. Iqbal, H., Campo, D., Baydoun, M., Marcenaro, L., Gomez, D.M., Regazzoni, C.: Clustering optimization for abnormality detection in semi-autonomous systems. In: 1st International Workshop on Multimodal Understanding and Learning for Embodied Applications, pp. 33–41 (2019)
13. Marin-Plaza, P.: Stereo vision-based local occupancy grid map for autonomous navigation in ros. In: 11th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications VISIGRAPP, vol. 2016 (2016)