2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Visual Forecasting as a Mid-level Representation for Avoidance

Hsuan-Kung Yang[1], Tsung-Chih Chiang[1*], Ting-Ru Liu[1*], Chun-Wei Huang[1*], Jou-Min Liu[1*], and Chun-Yi Lee[12]

***Abstract*— The challenge of navigation in environments with dynamic objects continues to be a central issue in the study of autonomous agents. While predictive methods hold promise, their reliance on precise state information makes them less practical for real-world implementation. This study presents visual forecasting as an innovative alternative. By introducing intuitive visual cues, this approach projects the future trajectories of dynamic objects to improve agent perception and enable anticipatory actions. Our research explores two distinct strategies for conveying predictive information through visual forecasting: (1) sequences of bounding boxes, and (2) augmented paths. To validate the proposed visual forecasting strategies, we initiate evaluations in simulated environments using the Unity engine and then extend these evaluations to real-world scenarios to assess both practicality and effectiveness. The results confirm the viability of visual forecasting as a promising solution for navigation and obstacle avoidance in dynamic environments.**

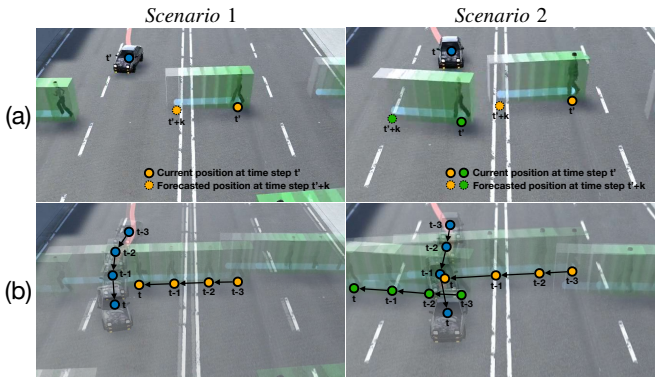

Fig. 1. The demonstrations of (a) the forecasted trajectories and (b) the agents interacting and avoiding pedestrians in the simulated environments.

I. INTRODUCTION

The challenge for autonomous agents to navigate and avoid obstacles in environments featuring dynamic objects presents a significant challenge and has become a central focus in the field. One prevailing direction to address this challenge involves utilizing predictive information to augment the performance of the agents [1]–[8]. Although predictive methods enhance performance in downstream tasks, they rely heavily on precise state information for both the agent and the environment. The acquisition of necessary data often demands significant effort, which becomes a major obstacle in deploying these methods for real-world navigation and obstacle avoidance scenarios. Furthermore, the complexity of interactions and dynamics in these environments complicates the agents' ability to interpret responses effectively and efficiently. As a result, these difficulties have prompted interests in exploring alternative approaches that rely on more direct, visually-oriented representational solutions that can offer an immediate and intuitive understanding of environmental dynamics. Within this context, we introduce the concept of visual forecasting as a compelling solution. This methodology employs intuitive visual cues to depict the future trajectories of dynamic objects and serves two purposes: it aids the agent's perception process and enhances its capacity for anticipatory action, which enables effective responses to forthcoming events. Moreover, it eliminates the need for precise environmental state acquisition, which offers promising avenues for implementation in real-world settings.

In light of the above, it becomes essential to identify an approach to incorporate visual forecasting information into the representations presented to an autonomous agent in a manner that is both effective and efficient. The task requires the translation of complex dynamic information into a visual format that is both digestible and comprehensive, which allows the agent to interpret and react promptly. Overly simplified or exceedingly complicated representations could result in inefficient processing or the omission of vital cues. A viable solution to this challenge may be grounded on the adoption of mid-level representations [9]–[13]. Each type of mid-level representation focuses on a specific abstract perspective of visual information. It offers the potential to achieve a balance between high-level abstract information and low-level sensory data, and provides potential for facilitating sim-to-real transfer. By utilizing mid-level representations, the visual forecasting strategies concerned in this study can encapsulate essential aspects of forecast information in a manner that is sufficiently comprehensive and computationally efficient. This study initiates an investigation into two viable strategies for visual forecasting through mid-level representations: (1) a sequence of bounding boxes to outline objects' positions over time, and (2) an augmented path to render the projected trajectory of movement. The selection of these strategies stems from their feasibility in real-world applications, which is supported by the advancements in object detection and tracking models [14]–[19], alongside the integration of mixed reality technologies [20]–[22]. Each strategy offers a distinct mechanism to integrate information into mid-level representations in order to convey future predictive information to the agent. Our objective is to investigate whether the modified mid-level representations can improve comprehension and offer feasible guidance to autonomous agents. In addition, we aim to explore their

* indicates equal contribution.
[1] Elsa Lab, National Tsing Hua University, Hsinchu City, Taiwan.
[2] Elsa Lab, National Taiwan University, Taipei City, Taiwan.

transferability to real scenarios to validate their effectiveness.

To validate the proposed visual forecasting strategies, we first utilize the Unity engine [23] to conduct a comprehensive set of evaluations in simulated environments. These environments support the generation of configurable dynamic objects, and offer complete customization capabilities to meet specific evaluation criteria. Our evaluation begins by examining the effectiveness of visual forecasting and then proceeds to analyze the underlying causes of the failure cases. Moreover, our evaluations examine the impact of forecasting quality, particularly through the use of the Constant Velocity Model (CVM) and Kalman Filter (KF), on the performance of downstream tasks. This analysis specifically targets object avoidance tasks executed by deep reinforcement learning (DRL) agents. To further validate our concepts and their applicability, this study incorporates real-world scenarios. Despite the inherent challenges of these scenarios, such as complexity and uncertainty, they offer valuable insights into the practicality of our forecasting schemes. Our analyses and findings from simulated and real-world settings validate the efficacy of visual forecasting in aiding autonomous agents. The contributions of this paper are summarized as follows:

- We introduce the concept of visual forecasting as a type of mid-level representation to present predictive information directly in the DRL agent's observations, which allows the agent to interpret and react promptly.
- We provide a comprehensive set of analyses on different types of visual forecasting representations, including bounding boxes and augmented paths, and validate the effectiveness in both simulated and real-world scenarios.
- We evaluate the impact of forecasting quality through CVM and KF on the performance of the DRL agents.

II. Preliminary

A. Virtual-to-Real Transfer via Mid-Level Representations

Mid-level representations serve as crucial abstract constructs that encapsulate a variety of physical or semantic aspects inherent in visual scenes. These domain-invariant properties, whether extracted or inferred, have demonstrated their significance across a broad spectrum of applications, and facilitate efficient information transfer from perception modules to control modules [9]–[12], [24]. These representations can take a variety of forms, such as depth maps, raw optical flow, semantic segmentation, etc. Each of these forms exhibits unique strengths and potential limitations depending on the specifics of the scenario [25]. Recent research [24] has introduced the concept of virtual guidance as a novel mid-level representation. This approach generates semantic segmentation-like virtual markers to direct agents along a predetermined path. This reveals the potential of adaptively modifying mid-level representations as a means to supply agents additional information to enhance their decision-making abilities. Recognizing the significance of mid-level representations is essential for the successful implementation of modular learning-based frameworks. As a result, this study aims to utilize the concept of mid-level representation and explore methods to represent forecasted trajectories of dynamic objects to facilitate decision-making of the agents.

B. From Forecasting to Action

The primary focus of several seminal works has predominantly been on the aspect of prediction, with less emphasis placed on utilizing these predictive models for subsequent downstream tasks [26]–[31]. Nonetheless, recent literature increasingly emphasizes the advantageous integration of forecasting models into various tasks. For instance, the study in [32] explored the domain of visual forecasting with action-conditioned predictions, which can be used for future action planning. Within the RL domain, PiSAC [33] highlighted the advantages of predictive information, and suggested that the incorporation of such information can expedite the learning process. In the context of navigation tasks, several works have made progress. Studies such as [6]–[8] have combined forecasting with model predictive control (MPC) to enhance performance. However, these studies often rely on the assumption that position data are readily available, and oftentimes utilize straightforward location information instead of richer high-dimensional RGB data. Deriving such positional information, however, typically necessitates additional sensors or sophisticated position estimation models. There have been research endeavors [34], [35] which have integrated RGB input with MPC and demonstrated the viability of RGB-based control tasks. Despite the feasibility of MPC for forecasting in a range of tasks, applying this technique to real-world visual navigation and obstacle avoidance remains challenging. One issue that warrants mention is the need for an accurate world model. Such models often face difficulties when dealing with high-dimensional state spaces inherent in visual inputs. It is important to clarify that this study aims to introduce methods for representing forecasted information on visual observations. Specifically, the focus is on presenting information related to dynamic entities in a manner that is both intuitive and easily comprehensible for agents. While it holds potential for integration with MPC, the objective of this paper remains distinct from MPC based methodologies.

C. Reinforcement Learning and Soft Actor-Critic (SAC)

In the context of reinforcement learning (RL), an agent interacts with an environment $\mathcal{E}$, which can be described using a Markov Decision Process (MDP). This process involves the agent observing a state s_t from a set of possible states $\mathcal{S}$ (i.e., the state space) at each timestep t. The agent then takes an action a_t from a set of possible actions $\mathcal{A}$ (i.e., the action space) based on its policy π, and receives a reward r_t from the $\mathcal{E}$. The goal of the agent is to learn an optimal policy π^* that maximizes the expected sum of discounted rewards $G_t = \mathbb{E}\big[\sum_{\tau=t}^{T} \gamma^{\tau-t} r_\tau\big]$, where γ represents the discount factor and T denotes the horizon of an episode [36], [37]. To promote the agent's exploratory actions, maximum entropy reinforcement learning [38] advocates for the optimization of a policy π that not only seeks to maximize expected returns but also encourages policy entropy, thus defined as $G_t = \mathbb{E}[\sum_{\tau=t}^{T} \gamma^{\tau-t}(r_\tau + \alpha\mathcal{H}(\pi))]$,

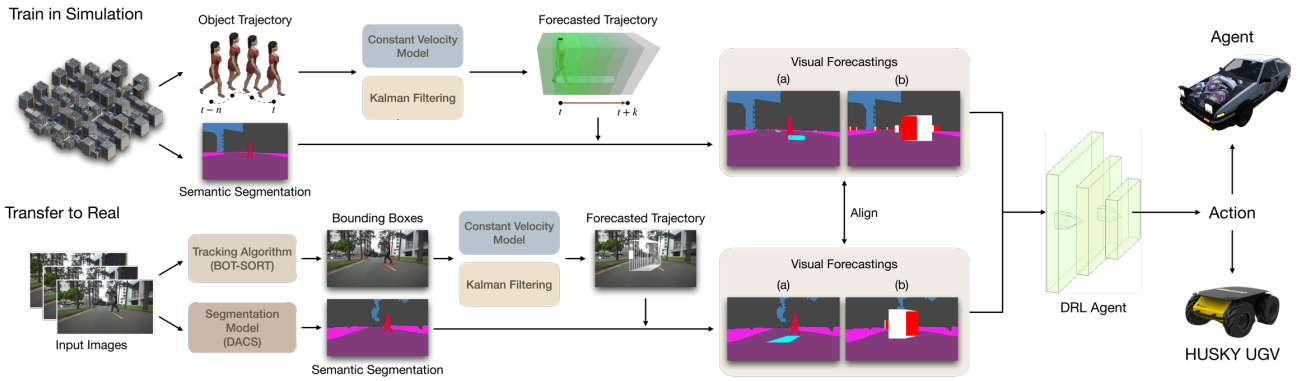

Fig. 2. An Overview of our framework.

where $\mathcal{H}(\pi) = \mathbb{E}[-\log \pi(\cdot|s_t)]$ is the entropy of the policy and α is a temperature parameter for adjusting the entropy's effect on the expected return. Soft Actor-Critic (SAC) [39]–[41] extends these concepts into deep reinforcement learning (DRL) by incorporating maximum entropy principles with deep neural networks (DNNs) to manage the complexities of high-dimensional state spaces. SAC employs a soft-Q function Q_θ and a stochastic policy π_ϕ, with θ and ϕ representing the DNN weights. This approach enhances the DRL agent's exploration capabilities and performance in both continuous [40], [41] and discrete action spaces [39].

III. Methodology

Visual forecasting operates as a new type of mid-level representation, which is designed to abstract and illustrate present forecasted information about moving objects. This approach enables agents to interpret the forecasted information through visual inputs. Such an approach carries multiple advantages. The primary benefit lies in facilitating intuitive comprehension. By rendering temporally forecasted information into visually understandable formats, visual forecasting simplifies the processing requirements for the agent. This is because the forecasted information is integrated with image observations, which reduces the agent's effort to process an additional modality (e.g., numerical vectors). Second, it enhances the agent's grasp of contextual information. While observations made by the agent typically contain information from only the current timestep, the modification of mid-level representation to incorporate visual forecasting allows the agent to access future forecasted information within its current observation. By enabling the agent to directly observe the forecasted trajectories of dynamic objects, it can more holistically perceive the environment and its dynamics, which in turn support the agent for more informed decision-making.

A. Overview of the Framework

Fig. 2 provides an overview of the framework, which is composed of two parts: (1) *Train in Simulation* and (2) *Transfer to Real*. In the *Train in Simulation* phase, each round of simulation produces a set of mid-level representations, including semantic segmentation, detected bounding boxes, and tracked trajectories. These representations form the basis

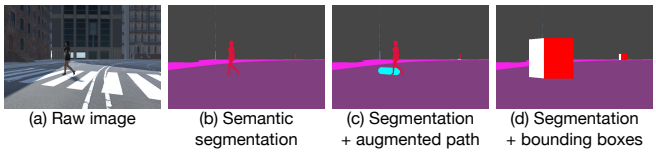

Fig. 3. Visualizations of different visual forecasting representations.

for estimating future trajectories, which can be derived from either a Constant Velocity Model (CVM) or the Kalman Filter (KF) [42]. For the *Transfer to Real* phase, input images undergo preprocessing via an object detection and tracking algorithm (e.g., BOT-SORT [15]) as well as a segmentation model (e.g., DACS [43]). This preprocessing generates mid-level representations that correspond to those used in simulation for predicting future trajectories. This work examines two visual forecasting representations, which are detailed in Section III-B. Either of these representation is rendered directly onto the semantic segmentation to form the agent's input observation. When transferring the trained DRL agents to real-world scenarios, the visual forecasting representation employed aligns with the one utilized in the simulated environments. Specifically, the configurations pertaining to visual forecasting, such as the colors and geometries superimposed onto the input observations, are preserved to ensure consistency. The trained DRL agent's model is fixed, eliminating the need for further fine-tuning when transferring from virtual to real-world environments.

B. Representations for Visual Forecasting

In this section, we discuss two visual forecasting strategies utilizing different perspectives on object motion prediction.

a) Bounding Box (BOX): One potential strategy for visual forecasting can be represented through a sequence of bounding boxes, denoted by $B_t, B_{t+1}, \ldots, B_{t+k}$, which indicate the predicted future locations of a specific target object at timesteps ranging from t to $t+k$. Specifically, each bounding box B_t is characterized by the tuple (x_t, y_t, w_t, h_t), where (x_t, y_t) denotes the coordinates of the upper-left corner of B_t, whereas w_t and h_t represent the width and height of B_t, respectively. The current bounding box B_t and the forecasted ones $B_{t+1}, \ldots B_{t+k}$ are colored differently to

allow the agent to identify them, as depicted in Fig. 3 (d).

b) Augmented Path (AP): The augmented path represents another strategy constructed by connecting coordinates from the bounding boxes B_t and B_{t+k}. Specifically, the coordinates $(x_t, y_t + h_t)$, $(x_t + w_t, y_t + h_t)$, $(x_{t+k}, y_{t+k} + h_{t+k})$, and $(x_{t+k} + w_{t+k}, y_{t+k} + h_{t+k})$ are linked to create an area that visualizes an object's forecasted trajectory. This strategy offers an intuitive depiction of the future path by extending the lower contours of the bounding boxes, and creates a visual pathway to indicate the anticipated object motion from t to $t+k$. This strategy is depicted in Fig. 3 (c).

C. Generation of Visual Forecasting

This section presents the generation workflow, which encompasses two distinct aspects: (1) the derivation of visual forecasting within the simulated environments, and (2) the formulation of visual forecasting in real-world scenarios.

1) Derivation of Visual Forecasting in Simulated Environments: Within our simulated environments, historical trajectories of moving objects, specifically pedestrians in this study, are directly obtainable through the Unity engine [23]. These trajectories include the complete history of positions and bounding boxes for the moving objects in world coordinates. Given a sequence of pedestrian positions and bounding boxes, either CVM or KF can be employed to estimate future positions and bounding boxes. These estimations then undergo post-processing and are represented via the visual forecasting representations, as discussed in Section III-B.

2) Formulation of Visual Forecasting in Real-world Scenarios: Distinct from the procedure discussed in Section III-C.1, generating visual forecasting in real-world scenarios presents a unique set of challenges. While techniques such as CVM and KF can still be employed to estimate future pedestrian locations, acquiring a complete historical trajectory for each pedestrian could be a complicated matter, especially for a moving camera. The difficulty arises primarily from the ego-centric motion induced by the moving camera. Such motion introduces ambiguity between the object's own motion and the observer's motion, which complicates the task of identifying the cause of bounding box movement. This ambiguity may lead to inaccuracies of state vectors used in KF, and impact the performance of tracking algorithms that rely on KF [44], [45]. To mitigate the effects of ego-centric motion, BOT-SORT [15], which incorporates camera-motion compensation, is utilized to deliver more stable historical trajectories. These trajectories, which comprise positions and bounding box coordinates, are utilized as inputs for either CVM or KF to derive forecasted bounding boxes.

IV. Experimental Results

The validations of our methodology are conducted in both simulated and real-world settings. In real-world environments, our model relies exclusively on a monocular camera for capturing RGB images, and no depth sensor is utilized.

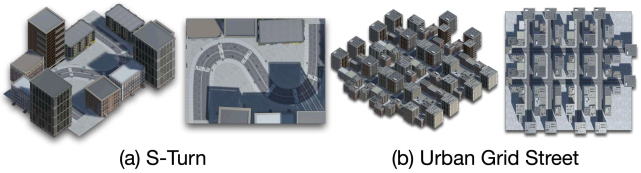

Fig. 4. An overview of the simulated environments utilized in our experiments. The *S-Turn* environment in (a) features an S-shaped path, while the *Urban Grid Street* scenario in (b) is a setting with eight intersections.

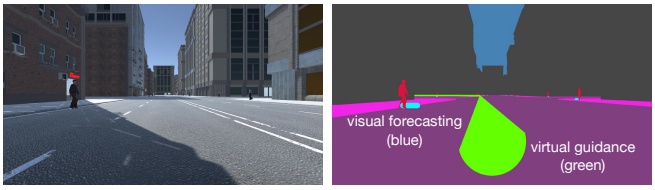

Fig. 5. Visualizations of virtual guidance [24] and visual forecasting.

A. Experimental Setup

1) Simulated Environment Setup: To evaluate the effectiveness of visual forecasting, we utilize two environments developed using the Unity engine [23] in our experiments: *S-Turn* [25] and *Urban Grid Street* [24]. Visual depictions of these environments are provided in Fig. 4. The simulated environments employed in our experiments are designed with configurable starting points and destinations, and incorporate dynamic objects (e.g., pedestrians) with customizable speeds. The objective of the agent is to navigate while avoiding both dynamic and static objects, based on the provided observations such as semantic segmentation and visual forecasting.

In the *S-Turn* environment, agents undergo a straightforward test, being trained and evaluated on identical route combinations, albeit with varied pedestrian speeds. The *Urban Grid Street* environment, on the other hand, offers a more rigorous testing scheme. Following the setups in [24], the routes are categorized into: (a) *seen* and (b) *unseen* routes. The *seen* routes scenario aims to evaluate the agents' capability to navigate to their destination without encountering obstacles, using the same combinations of starting points and destinations as those in the training phase. In contrast, the *unseen* routes challenge agents with novel combinations of start and end points. In total, 89 routes are used for training the agents, while a distinct set of four routes are selected for the evaluation phase. In both environments, pedestrians are configured with speeds of 0.6 to $1.2\,\mathrm{m/s}$ during training, and a wider range of 0.3 to $1.5\,\mathrm{m/s}$ for the evaluation phase.

2) Agent Setup: In our experiments, the DRL agent is implemented as a deep neural network (DNN) trained using the Soft Actor-Critic (SAC) algorithm [39], [46]. The agent's observation comprises three stacked semantic segmentation frames, each with dimensions 84×180. These frames may be presented with or without the incorporation of virtual guidance. The agent operates within an action space defined by a set $\mathcal{A}$. This set comprises two primary actions: **NOOP** and **TURN**(α). Under the **NOOP** action, the agent maintains

TABLE I

A QUANTITATIVE COMPARISON FOR DEMONSTRATING THE BENEFICIAL IMPACTS OF THE PRPOSED VISUAL FORECASTING SCHEMES.

Approaches	Visual Forecasting	Success Rate Weighted by Path Length (SPL)						Success Rate					
		[0.3 m/s, 0.6 m/s]		[0.6 m/s, 1.2 m/s]		[1.2 m/s, 1.5 m/s]		[0.3 m/s, 0.6 m/s]		[0.6 m/s, 1.2 m/s]		[1.2 m/s, 1.5 m/s]	
Seg (box)	✗	82.14 %	(↑ 2.32 %)	68.47 %	(↑ 11.77 %)	47.55 %	(↑ 25.89 %)	88.25 %	(↑ 4.27 %)	74.27 %	(↑ 14.16 %)	51.91 %	(↑ 29.59 %)
Seg (box) + BOX	✓	83.86 %		81.67 %		75.83 %		92.03 %		89.67 %		83.63 %	
Seg	✗	79.94 %	(↑ 6.20 %)	62.28 %	(↑ 19.61 %)	39.88 %	(↑ 35.12 %)	85.41 %	(↑ 6.33 %)	67.29 %	(↑ 20.45 %)	43.29 %	(↑ 37.63 %)
Seg + AP	✓	86.14 %		81.89 %		75.00 %		91.74 %		87.74 %		80.92 %	

its current directional orientation and travels along a straight trajectory. On the other hand, the **TURN**(α) action allows the agent to incrementally adjust its orientation based on the value α. The sign of α is essential: negative values induce a leftward adjustment, while positive values prompt a rightward shift. This highlights the agent's ability to navigate and turn in a non-binary fashion. The angular velocity ω, influenced by the continuous adjustments to α, is given by:

$$\omega \mathrel{+}= \alpha \times \kappa \times \Delta t, \tag{1}$$

where Δt represents the time interval, and κ defines the steering sensitivity. In our settings, α has a standard value of $35\,^{\circ}/\mathrm{s}^2$, and κ is set to two. Rather than abrupt binary switches in direction, our agent's trajectory evolves based on the cumulative impact of successive α adjustments. While the agent maintains a consistent velocity v of $6\,\mathrm{m/s}$, its direction continuously experiences refined and gradual modifications. It receives a reward of 10.0 when reaching the destination, and a penalty of -10.0 for colliding with an obstacle, venturing outside the boundaries, or exceeding the time limit.

3) Evaluation Metrics:

a) Success rate: The success rate is a metric for assessing the proficiency of an agent in reaching the destination.

b) Success rate weighted by path length (SPL): The SPL metric evaluates the agent's navigational performance by accounting for both the success in reaching the destination and the efficiency of the selected trajectory [47]. This comparison ensures a consideration of both navigational success and efficiency. SPL is mathematically represented as follows:

$$\frac{1}{N}\sum_{i=1}^{N}\frac{l_i}{\max(l_i, p_i)}, \tag{2}$$

where l_i represents the shortest path distance from the agent's starting position to the goal for episode i, and p_i denotes the length of the path actually taken by the agent in that episode.

c) Collision rate and out-of-bound (OOB) rate: The collision and the out-of-bound rate are utilized as our metrics to analyze the causes of failure cases. Each failure case can be attributed to one of two possible causes: (a) *out-of-bound*, which indicates the agent has ventured into prohibited regions such as the sidewalk, and (b) *collision*, which represents instances where the agent collides with obstacles.

d) Final displacement error (FDE) and average displacement error (ADE): ADE is defined as the mean Euclidean distance between the predicted and the ground-truth future positions. FDE, on the other hand, is defined in a similar manner but is calculated only for the final timestep.

4) Hyperparameters: In this study, the input resolution for object detection and segmentation models are set to 360×640 and 720×1280, respectively. We forecast bounding boxes for the future five time intervals, where each interval comprises four skipped frames. For each experiment, we employ three independent and identically distributed (i.i.d.) random seeds.

5) Baseline Representations: In this study, two baseline representations are considered for comparison: *Seg* and *Seg (box)*. The *Seg (box)* representation extends the pedestrian segment to its bounding box, to enable a fair comparison with *seg (box) + BOX* to evaluate the efficacy of visual forecasting. This representation is particularly relevant given the enlarged coverage that bounding boxes provide for individual pedestrians. While *Seg* captures the detailed contours of pedestrians, neither *BOX* nor *Seg (box)* offer this level of granularity.

6) Hardware Configuration: Our real-world experiments were carried out on a laptop equipped with an NVIDIA GeForce RTX 4080 Mobile GPU and an Intel Core i9 CPU. The robot platform is a ClearPath Husky Unmanned Ground Vehicle (UGV). The entire pipeline, encompassing object detection, tracking, and segmentation, achieved an inference time of 0.03 seconds (i.e., $\sim$ 33 frames per second (FPS)).

B. Validation of the Effectiveness of Visual Forecasting

In this section, we focus on validating the effectiveness of visual forecasting, with a particular emphasis on its role in enhancing the performance of DRL agents through CVM. Our evaluations are conducted using the *S-Turn* environment, as the primary objective is to assess the agent's object avoidance capabilities. Evaluations of the other forecasting strategies will be discussed in the next section. Table I presents the evaluation results. The visualized demonstrations of the forecasted trajectories, along with the agents' interactions, are illustrated in Fig. 1 and Fig. 6. The results suggest that the agents equipped with visual forecasting consistently outperform those without this feature in terms of both the *SPL* and *success rate* metrics under two different visual forecasting configurations. These findings indicate that visual forecasting is able to provide informative foresight into the potential trajectories of dynamic objects, and enables the agents to plan ahead as well as avoiding potential collisions.

To further interpret the implications of visual forecasting, we undertake an analytical examination of the failure cases to identify their root causes. As evidenced in Table II, the agents trained with visual forecasting exhibit a lower *collision rate* in comparison to their counterparts trained without visual forecasting. This suggests the efficacy of visual forecasting in

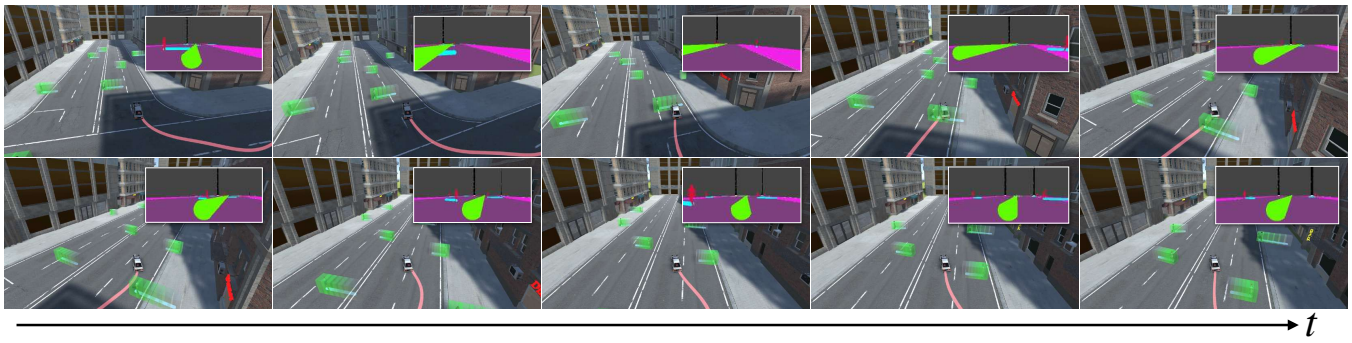

Fig. 6. Demonstrations of agents' trajectories and actions, along with the rendered virtual guidance signals in simulated environments.

TABLE II
AN ANALYSIS OF FAILURE CASES.

Approaches	Visual Forecasting	Causes of the Failure Cases	
		Collision Rate (↓)	*Out-Of-Bound Rate* (↓)
Seg (box)	✗	24.19 %	4.33 %
Seg (box) + BOX	✓	7.34 %	4.22 %
Seg	✗	29.40 %	5.27 %
Seg + AP	✓	9.73 %	3.48 %

TABLE III
AN ANALYIS OF THE IMPACT OF THE FORECASTING QUALITY ON THE PERFORMANCE OF DRL AGENT.

Visual Forecasting	Algorithm	FDE (↓)	ADE (↓)	SPL (↑)	Success Rate (↑)
BOX	✗	—	—	66.05 %	71.48 %
	CVM	0.471	0.283	80.45 %	88.44 %
	KF	0.456	0.340	82.71 %	89.22 %
	GT	—	—	83.85 %	90.83 %
AP	✗	—	—	60.70 %	65.33 %
	CVM	0.471	0.283	81.01 %	86.80 %
	KF	0.456	0.340	81.63 %	87.27 %
	GT	—	—	83.53 %	87.90 %

aiding the agents to interpret the motions of dynamic objects.

C. Quality of Forecasting: Is Constant Velocity Sufficient?

To demonstrate the significance of visual forecasting, the previous section highlighted that the agents, even when guided by the most straightforward CVM forecasting approach, outperform those without visual forecasting. While the accuracy of visual forecasting estimations might be pertinent to certain downstream tasks and high forecasting accuracy is often desired, its presence does not always signify improved outcomes in navigation and obstacle avoidance. As a result, this section investigates the relationship between forecasting accuracy and the agent's capabilities in navigation and object avoidance. For this examination, we utilize both CVM and KF as forecasting mechanisms, and compare their outcomes with the Ground Truth (GT) forecasting as the benchmark for ideal future location predictions characterized by zero FDE and ADE errors. Our evaluation is performed on the *S-Turn* environment and encompasses two distinct perspectives: (a) the precision of forecasting, measured by FDE and ADE over five steps across 100 pedestrian samples, and (b) the proficiency of the agent, reflected through the metrics of *SPL* and *success rate*. The results are presented in Table III. It can be observed that the forecasting accuracy of the CVM closely approximates that of the KF. Both models are able to deliver satisfactory results, with only slight margins compared to the perfect predictions of the GT forecasting. This finding aligns with the previous studies in [48]–[50]. These experimental findings therefore suggest the efficacy of even a CVM in producing reliable forecasts.

Table III provides further validation regarding the influence of forecasting quality on an agent's ability in object avoidance. As indicated by the SPL metric in Table III, even basic models such as CVM or KF offer beneficial insights to DRL agents in object avoidance tasks. It is worth noting that the agents trained with forecasts from CVM and KF display performance levels nearing those trained with the GT's impeccable visual forecasting. The explanation to this observation could be that the primary advantage of visual forecasting lies in its capability to offer an intuitive depiction of general trajectories. Even without intricate details, these visual indications prove to be highly effective in directing agents. The combined simplicity and efficacy of CVM suggest its potential for adaptation and practical use in real-world situations, given its low computational requirements.

D. Compatibility of Visual Forecasting with Navigation

In this section, we investigate the interplay between visual forecasting and the virtual guidance-based navigation technique introduced in [24]. Building on the premise established in the previous sections that visual forecasting can assist agents, the *Urban Grid Street* environment is selected for this examination. This environment serves as a more generalized setting and allows us to adopt a diverse combination of routes to assess the robustness and adaptability of the forecasting models. The integration of these two methods is depicted in Fig. 5. The evaluation results are presented in Table IV. It can be observed that the agents trained with visual forecasting consistently outperform those without it, as evidenced by the *SPL* and *success rate* metrics. This not only highlights the complementary nature of visual forecasting and virtual navigation schemes, but also reveals the DRL agents' ability to comprehend information from these two different methods.

TABLE IV

THE RESULTS OF VISUAL FORECASTING WITH VIRTUAL NAVIGATION.

Scenario	Approaches	SPL	Success Rate
seen	*Seg*	65.60 %	65.78 %
	Seg + AP	77.92 %	78.19 %
unseen	*Seg*	47.06 %	47.50 %
	Seg + AP	63.30 %	63.83 %

TABLE V

COMPARISON OF UTILIZING 2D AND 3D BOUNDING BOXES FOR PERFORMING VISUAL FORECASTING.

Approaches	Require 3D?	SPL	Success Rate
Seg (box)	—	66.05 %	71.48 %
Seg (box) + BOX (2D)	✗	71.65 %	77.33 %
Seg (box) + BOX (3D)	✓	79.38 %	87.48 %
Seg	—	60.70 %	65.33 %
Seg + AP (2D)	✗	71.89 %	78.00 %
Seg + AP (3D)	✓	81.01 %	86.80 %

TABLE VI

THE EVALUATION RESULTS IN THE REAL-WORLD ENVIRONMENT.

Approaches	Completion Rate	Success Rate
Seg (box)	20.00 %	5.00 %
Seg (box) + BOX	58.33 %	25.00 %
Seg	31.67 %	5.00 %
Seg + AP	73.33 %	60.00 %

Fig. 7. Demonstration of the agent's performance in real-world scenarios.

E. Real World Transferring

In this section, we evaluate the trained agent's performance in real-world scenarios to understand the potential of visual forecasting as a mid-level representation and its efficacy in aiding agents with object avoidance tasks. A consideration prior to this transition is the agent's reliance on 2D information extracted from RGB images, as discussed in Section III-C.2, especially in the absence of precise 3D information about objects. To circumvent these limitations, we introduce a modified approach that relies solely on predicted 2D bounding boxes for visual forecasting. The results, presented in Table V, demonstrate the adaptability of our method in real-world settings. It is evident that using even just 2D bounding boxes can improve the agent's performance compared to not using visual forecasting at all, although the effectiveness is somewhat less than when utilizing 3D bounding boxes. Given that 3D prediction models could demand more computational resources and are not the primary focus of this paper, these findings serve as a preliminary indication that satisfactory performance can be achieved even with 2D bounding boxes.

The results of our real-world experiments are presented in Table IV-C. The agents are evaluated both with and without the incorporation of visual forecasting. Each experiment is conducted over twenty independent runs, and the task involves guiding the agent to a predetermined destination while avoiding collisions with three moving pedestrians. We use the *completion rate* metric to evaluate the agent's ability to dodge pedestrians successfully. It can be observed that the agents equipped with visual forecasting (i.e., *Seg + AP* and *Seg (box) + BOX*) outperform those without forecasting. Fig. 7 further illustrates this by showcasing two examples of the agent behavior using the *Seg + AP* forecasting scheme. In both instances, the agents appear to take into account the future positions of the pedestrians when making decisions.

V. CONCLUSION

In this paper, we address the challenges of navigation and obstacle avoidance for autonomous agents in dynamic environments. We introduce visual forecasting as an innovative strategy that utilizes intuitive visual cues to predict the future paths of dynamic objects. Specifically, our visual forecasting approach acts as a type of mid-level representation, which seamlessly incorporates predictive information into the observations received by DRL agents. This integration ensures the agent can interpret the information efficiently. In this study, we explore two distinct strategies for conveying predictive information through visual forecasting: (1) sequences of bounding boxes, and (2) augmented paths. To validate our approach, we conducted experiments in both simulated and real-world settings. Our experimental investigations, carried out in both simulated and real-world settings, confirm the efficacy and practical applicability of visual forecasting in facilitating obstacle avoidance task for autonomous agents.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support from the National Science and Technology Council (NSTC) in Taiwan under grant numbers MOST 111-2223-E-002-011-MY3, NSTC 113-2221-E-002-212-MY3, and NSTC 113-2640-E-002-003. The authors would like to express their appreciation for the donation of the GPUs from NVIDIA Corporation and NVIDIA AI Technology Center (NVAITC) used in this work. Furthermore, the authors extend their gratitude to the National Center for High-Performance Computing (NCHC) for providing the necessary computational and storage resources.

REFERENCES

[1] Z. Liu, Q. Liu, W. Xu, Z. Liu, Z. Zhou, and J. Chen, "Deep learning-based human motion prediction considering context awareness for human-robot collaboration in manufacturing," *Procedia CIRP*, vol. 83, pp. 272–278, 2019.

[2] S. Poddar, C. Mavrogiannis, and S. S. Srinivasa, "From crowd motion prediction to robot navigation in crowds," *arXiv preprint arXiv:2303.01424*, 2023.

[3] A. Wang, C. Mavrogiannis, and A. Steinfeld, "Group-based motion prediction for navigation in crowded environments," in *Proc. Conf. on Robot Learning (CoRL)*, 2022, pp. 871–882.

[4] C. Park, J. Ondřej, M. Gilbert, K. Freeman, and C. O'Sullivan, "Hi robot: Human intention-aware robot planning for safe and efficient navigation in crowds," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2016, pp. 3320–3326.

[5] Z. Chen, C. Song, Y. Yang, B. Zhao, Y. Hu, S. B. Liu, and J. Zhang, "Robot navigation based on human trajectory prediction and multiple travel modes," *Applied Sciences*, 2018.

[6] T. Fraichard and V. Levesy, "From crowd simulation to robot navigation in crowds," *IEEE Robotics and Automation Letters*, 2020.

[7] A. Feher, S. Aradi, and T. Becsi, "Online trajectory planning with reinforcement learning for pedestrian avoidance," *Electronics*, 2022.

[8] C. Mavrogiannis, K. Balasubramanian, S. Poddar, A. Gandra, and S. S. Srinivasa, "Winding through: Crowd navigation via topological invariance," *IEEE Robotics and Automation Letters*, 2021.

[9] Z.-W. Hong, Y.-M. Chen, H.-K. Yang, S.-Y. Su, T.-Y. Shann, Y.-H. Chang, B. H.-L. Ho, C.-C. Tu, T.-C. Hsiao, H.-W. Hsiao, S.-P. Lai, Y.-C. Chang, and C.-Y. Lee, "Virtual-to-real: Learning to control in visual semantic segmentation," in *Proc. Int. Joint Conf. on Artificial Intelligence (IJCAI)*, 2018, pp. 4912–4920.

[10] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in *SSCI*, 2020.

[11] Y.-C. Lin, A. Zeng, S. Song, P. Isola, and T.-Y. Lin, "Learning to see before learning to act: Visual pre-training for manipulation," in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, 2020, pp. 7286–7293.

[12] B. Chen*, A. Sax*, F. Lewis, S. Savarese, A. Zamir, J. Malik, and L. Pinto, "Robust policies via mid-level visual representations: An experimental study in manipulation and navigation," in *4th Annual Conference on Robot Learning, CoRL 2020*, ser. Proceedings of Machine Learning Research. PMLR, 2020.

[13] A. Sax, B. Emi, A. R. Zamir, L. J. Guibas, S. Savarese, and J. Malik, "Mid-level visual representations improve generalization and sample efficiency for learning visuomotor policies." 2018.

[14] Y. Zhang, P. Sun, Y. Jiang, D. Yu, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," in *Proc. European Conf. on Computer Vision (ECCV)*, 2021.

[15] N. Aharon, R. Orfaig, and B.-Z. Bobrovsky, "Bot-sort: Robust associations multi-pedestrian tracking," *arXiv preprint arXiv:2206.14651*, 2022.

[16] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023, pp. 7464–7475.

[17] T. Cheng, L. Song, Y. Ge, W. Liu, X. Wang, and Y. Shan, "Yolo-world: Real-time open-vocabulary object detection," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[18] Y. Wang, J.-W. Hsieh, P.-Y. Chen, and M.-C. Chang, "Smiletrack: Similarity learning for multiple object tracking," *ArXiv*, vol. abs/2211.08824, 2022.

[19] J. Yang, M. Gao, Z. Li, S. Gao, F. Wang, and F. Zheng, "Track anything: Segment anything meets videos," 2023.

[20] M. Szalai, B. Varga, T. Tettamanti, and V. Tihanyi, "Mixed reality test environment for autonomous cars using unity 3d and sumo," 2020.

[21] H. B. Mohammadi, M.-A. Zamani, M. Kerzel, and S. Wermter, "Mixed-reality deep reinforcement learning for a reach-to-grasp task," in *Proc. Int. Conf. on Artificial Neural Networks*, 2019.

[22] M. Quinlan, T.-C. Au, J. Zhu, N. Stiurca, and P. Stone, "Bringing simulation to life: A mixed reality autonomous intersection," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2010, pp. 6083–6088.

[23] Unity Technologies, "Unity engine," https://unity.com.

[24] H.-K. Yang *et al.*, "Vision based virtual guidance for navigation," *arXiv preprint arXiv:2303.02731*, 2023.

[25] H.-K. Yang, T.-C. Hsiao, T.-H. Liao, H.-S. Liu, L.-Y. Tsao, T.-W. Wang, S.-Y. Yang, Y.-W. Chen, H.-R. Liao, and C.-Y. Lee, "Investigation of factorized optical flows as mid-level representations," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, 2022, pp. 746–753.

[26] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[27] T. Salzmann, B. Ivanovic, P. Chakravarty, and M. Pavone, "Trajectron++: Dynamically-feasible trajectory forecasting with heterogeneous data," in *Proc. European Conf. on Computer Vision (ECCV)*, 2020.

[28] K. Mangalam, E. Adeli, K.-H. Lee, A. Gaidon, and J. C. Niebles, "Disentangling human dynamics for pedestrian locomotion forecasting with noisy supervision," in *Proc. IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2020.

[29] Y. Yuan, X. Weng, Y. Ou, and K. M. Kitani, "Agentformer: Agent-aware transformers for socio-temporal multi-agent forecasting," in *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, October 2021, pp. 9813–9823.

[30] B. Pang, T. Zhao, X. Xie, and Y. N. Wu, "Trajectory prediction with latent belief energy-based model," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 11 814–11 824.

[31] N. Shafiee, T. Padir, and E. Elhamifar, "Introvert: Human trajectory prediction via conditional 3d attention," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[32] C. Finn, I. Goodfellow, and S. Levine, "Unsupervised learning for physical interaction through video prediction," in *Proc. Conf. on Neural Information Processing Systems (NeurIPS)*, 2016.

[33] K.-H. Lee *et al.*, "Predictive information accelerates learning in rl," in *Proc. Conf. on Neural Information Processing Systems (NeurIPS)*, 2020.

[34] F. Ebert, C. Finn, S. Dasari, A. Xie, A. Lee, and S. Levine, "Visual foresight: Model-based deep reinforcement learning for vision-based robotic control," *arXiv preprint arXiv:1812.00568*, 2018.

[35] K.-H. Zeng, R. Mottaghi, L. Weihs, and A. Farhadi, "Visual reaction: Learning to play catch with your drone," in *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[36] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, 1st ed. Cambridge, MA, USA: MIT Press, 1998.

[37] R. S. Sutton, D. Precup, and S. Singh, "Between MDPs and semi-MDPs: A framework for temporal abstraction in reinforcement learning," *Artificial Intelligence*, vol. 112, no. 1-2, pp. 181–211, 1999.

[38] T. Haarnoja, H. Tang, P. Abbeel, and S. Levine, "Reinforcement learning with deep energy-based policies," in *Proc. Int. Conf. on Machine Learning (ICML)*, 2017.

[39] P. Christodoulou, "Soft actor-critic for discrete action settings," *arXiv preprint arXiv:1910.07207*, 2019.

[40] T. Haarnoja, A. Zhou, K. Hartikainen *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2019.

[41] O. Delalleau *et al.*, "Discrete and continuous action representation for practical rl in video games," *arXiv preprint arXiv:1912.11077*, 2019.

[42] "A new approach to linear filtering and prediction problems," vol. 82, 1960, pp. 35–45.

[43] W. Tranheden, V. Olsson, J. Pinto, and L. Svensson, "Dacs: Domain adaptation via cross-domain mixed sampling," in *Proc. IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2021.

[44] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. of IEEE Int. Conf. on Image Processing (ICIP)*, 2017, pp. 3645–3649.

[45] Y. Zhang, P. Sun, Y. Jiang, D. Yu, F. Weng, Z. Yuan, P. Luo, W. Liu, and X. Wang, "Bytetrack: Multi-object tracking by associating every detection box," 2022.

[46] T. Haarnoja *et al.*, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *ICML*, 2018.

[47] P. Anderson, A. X. Chang, D. S. Chaplot, A. Dosovitskiy, S. Gupta, V. Koltun, J. Kosecka, J. Malik, R. Mottaghi, M. Savva, and A. R. Zamir, "On evaluation of embodied navigation agents," *arXiv preprint arXiv:1807.06757*.

[48] C. Schöller, V. Aravantinos, F. S. Lay, and A. Knoll, "What the constant velocity model can teach us about pedestrian motion prediction," *IEEE Robotics and Automation Letters*, vol. 5, pp. 1696–1703, 2019.

[49] H. Wu, T. Phong, C. Yu, P. Cai, S. Zheng, and D. Hsu, "What truly matters in trajectory prediction for autonomous driving?" *arXiv preprint arXiv:2306.15136*, 2023.

[50] N. Uhlemann, F. Fent, and M. Lienkamp, "Evaluating pedestrian trajectory prediction methods for the application in autonomous driving," *arXiv preprint arXiv:2306.15136*, 2023.