ERA, 31(2): 1134–1146.
DOI: 10.3934/era.2023057
Received: 08 October 2022
Revised: 07 December 2022
Accepted: 11 December 2022
Published: 20 December 2022

http://www.aimspress.com/journal/era

Research Article

CrowdGAIL: A spatiotemporal aware method for agent navigation

Longchao Da and Hua Wei*

Department of Informatics, New Jersey Institute of Technology, Newark, New Jersey 07102, USA

* **Correspondence:** Email: hua.wei@njit.edu; Tel: +19735965289.

Abstract: Agent navigation has been a crucial task in today's service and automated factories. Many efforts are to set specific rules for agents in a certain scenario to regulate the agent's behaviors. However, not all situations could be in advance considered, which might lead to terrible performance in a real-world application. In this paper, we propose CrowdGAIL, a method to learn from expert behaviors as an instructing policy, can train most 'human-like' agents in navigation problems without manually setting any reward function or beforehand regulations. First, the proposed model structure is based on generative adversarial imitation learning (GAIL), which imitates how humans take actions and move toward the target to a maximum extent, and by comparison, we prove the advantage of proximal policy optimization (PPO) to trust region policy optimization, thus, GAIL-PPO is what we base. Second, we design a special Sequential DemoBuffer compatible with the inner long short-term memory structure to apply spatiotemporal instruction on the agent's next step. Third, the paper demonstrates the potential of the model with an integrated social manner in a multi-agent scenario by considering human collision avoidance as well as social comfort distance. At last, experiments on the generated dataset from CrowdNav verify how close our model would act like a human being in the trajectory aspect and also how it could guide the multi-agents by avoiding any collision. Under the same evaluation metrics, CrowdGAIL shows better results compared with classic Social-GAN.

Keywords: GAIL; social awareness; agent navigation

1. Introduction

Human-agent interactions are getting more frequent along with the booming robotic industry. Robots are appearing on many occasions like in hotels, restaurants and probably transportation, in the future. However, it is common sense that current robots are still acting clumsily. Especially when an agent encounters an obstacle on its original planned navigation route, it usually vibrates or stops, spending some time re-profiling another choice and then moving on. This is how traditional agent navigation is acting. Just imagine that, if a transporting robot is hesitating its way while some traffic is

passing by, it will be extremely dangerous. In order to train more 'human-like' agents, many attempts have been tried as shown below.

The navigation tasks are tightly connected with pedestrian interaction issues. To make a robot better navigate, by helping it model the interactive scenarios more effectively could be a sound voice. Based on this, the Social Force Model [1] was proposed and enlightened later work [2–4]. However, this method is usually regarded as a fixed decision policy with manually set regulations. Next, with the development of reliable sensors, a sensor-informed goal navigation method was utilized to assist robots to keep active rather than getting lost in dense crowd situations [5]. However, these still cannot act like humans in some sharp turns or crossing scenes. Some scholars attribute the solution to this problem as a better route planning algorithm [6], some trajectory smoothing technologies are proposed to help robots pursue more feasible route planning, but this method mainly tends to solve the fixed obstacle troubles, and usually fails when applied to human dynamic changing behaviors. Our work mainly focuses on mimicking human social behaviors from expert demonstration to training a learned robot more hominoid in navigating. And this 'human-like' not only refers to a human-like trajectory but social awareness in terms of considering neighbors' relative positions to avoid collision and maintain a certain comfortable distance. To address the 'human-like' navigation challenge, and as a result of being inspired by the generative adversarial imitation learning algorithm, we proposed innovation to directly learn from the expert's behavior policy. As the original GAIL algorithm was introduced based on TRPO [7], we further verified the model's learning effectiveness by using PPO [8] in policy optimization through a comparison experiment under the metrics ADE and FDE [9]. By using GAIL-PPO as our base model, we discover the problem of agents could lose directions when there is a crossing scene, after analyzing the possible causes, we extend the model into GAIL-PPO-Padding-LSTM (or we call it MemoryGAIL), and design its sequential DemoBuffer to provide a compatible learning structure. Then, at last, we integrate a special social awareness feature extraction part to better mimic human social manners. In conclusion, the contributions to this paper are:

- First, we verify that GAIL-PPO outperforms GAIL-TRPO in our task with detailed analysis.
- Second, we found the best structure to integrate the spatial feature into the temporal observations using LSTM and a specially designed sequential demobuffer, and we prove the model's accurate imitation result.
- Third, we design a socially aware part and our trained model could capture the social manners to avoid collision with a certain social distance when navigating.
- Finally, we conduct comparison experiments and an ablation study to analyze each part's functioning of CrowdGAIL.

2. Related work

There have been some related attempts to solve robotics navigation problems. The first type of work is based on hand-crafted interaction models. The social force model [1] has inspired other methods by modeling the common behavior manners of pedestrians at intersections from aspects: destinations, other agents' influences, other agents' attractions, and the fluctuations. Based on that, some predicting methods grew [10–12]. They are trying to solve the problem by conducting accurate trajectory predictions first or navigating based on the predicted results.

Another direction leads researchers to learn from expert demonstrations, which is known as imitation learning (IL) [13]: given an expert experience dataset, train an agent to learn from an expert on their decision policies behind the behaviors. And the generative adversarial network (GAN) [14] inspired the domain with game theory: the generator competes with the discriminator to learn a trade-off, eventually maximizing the generator's performance. A lot of work achieves success by applying a GAN to the imitation learning process, such as [15–17].

Reinforcement Learning (RL), as a recently prevalent machine learning method, is showing potential in sequential decision-making tasks. With the help of RL, using the Markov decision model to solve continuous programming problems has been a trend. In [18], the author proposed a visualized simulator depicting the commonly seen robot and human encountering scenes. Also, by fixing the human policy as ORCA [19], they trained a well-performed robot agent who could navigate through the crowds and successfully get to the destinations. However, the training of RL models is a quite time-consuming task, and sometimes it would result in global optimal solutions, and then it will need resetting at some of the environment settings or reward functions. Due to the difficulty of defining reward functions in many real-world scenarios, inverse reinforcement learning has been proposed, and learning an internal and potential reward function first, from the demonstration data, and then using the learned function to instruct the next step of the classic RL process. Doing so is taken as a more probable way to solve real-world problems. And after that, generative adversarial imitation learning [20] was proved to get through from IRL to IL [21], and it is shown to be a more effective way of conducting IRL [22].

3. Methods

*3.1. **Problem definition with Markov decision process***

In our setting, the raw observation features item of an agent at time t_i is a tuple $\langle t_i, x_i, y_i, x_g, y_g\rangle$, then, We define a Markov decision process as a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma\rangle$, where an agent interacts with the environment through a sequence of observations, actions and reward signals. Because our model is set up in three steps, so gradually, the observation space is getting complicated, shifting from a single agent's present feature to its n steps sequential states, and then, a perception of the scene's neighbors' visible states. And each agent would execute an action $a_t \in \mathcal{A}$ at time step t from its state space $s_t \in \mathcal{S}$, according to its learned policy $\pi : \mathcal{S} \to \mathcal{A}$, applying this to the navigation problem, we define the continuous action space as $\langle v_x, v_y\rangle$, signifying the speed change in the X direction and Y direction, [-1, 1]. The model then receives a reward signal $r_t : \mathcal{S} \to \mathbb{R}$ from the discriminator and transits to the next state s_{t+1} according to the environment's dynamics. We use π_E to denote the expert policy.

This observation space and the work's structure are finally settled through three steps. For each step, we are trying to solve different problems. 1). At step one, we are designing a basic model to compare traditional GAIL behavior by respectively using PPO and TRPO optimization policies, so for the state space, it only considers its current position and transit to the next state based on the actions v_x and v_y. Its observation is $\langle t_i, x_i, y_i, x_g, y_g\rangle$. 2). at step two, we try to solve the agent lost direction problem, so based on our method, we need to memorize the agent's state window with a certain fixed window size: obs length n. So here in the LSTM-GAIL, the observation will be $\langle t_{i_1}, x_{i_1}, y_{i_1}, x_g, y_g, t_{i_2}, x_{i_2}, y_{i_2}, x_g, y_g ..... t_{i_n}, x_{i_n}, y_{i_n}, x_g, y_g\rangle$. The window will be initialized at the very beginning of each round, and its structure is in a padding way, which means that when at the step1,

there is no other historical information, so the value will be set as 0s. 3). At step three, we are applying the CrowdGAIL in a social scene and the scene will include a certain number of robots, also taken as an individual agent, here in our settings, we have 5 agents in total. So we eventually set the space as a joint state space, where the center agent will carry its own sequential observations and will also add its neighbors' current positions, for example. When there exist 4 neighbor agents, an agent's observation will be expanded to its original sequence add: $\langle x_{i_1}, y_{i_1}, x_{i_2}, y_{i_2}, x_{i_3}, y_{i_3}, x_{i_4}, y_{i_4}\rangle$.

3.2. Generative adversarial imitation learning

Inspired by [14], GAIL was proposed [20], which surpasses the intermediate step of learning a reward function, but it can directly learn a policy from expert demos. In the GAIL model, the generator π_θ is to generate state-action ($\mathcal{S} \times \mathcal{A}$) pairs matching that from the expert distributions, while the discriminator D_ω learns to tell the generated policy π_θ(θ denotes the parameters of the generator of the GAIL model) apart from the expert policy π_E. The objective of GAIL is to optimize the function below ($H(\pi)$ represents the causal entropy of the policy):

$$\mathbb{E}_{\pi_\theta}[\log(D(s,a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] - \lambda H(\pi_\theta) \tag{3.1}$$

Following this objective, the learning procedure of GAIL updating the parameters ω of the discriminator D_ω to maximize Eq (3.1) and performing trust region policy optimization (TRPO) to minimize Eq (3.1) with respect to θ, which parameterizes the policy generator π_θ. Here, the discrimination scores of the generated samples are regarded as costs (can be viewed as the negative counterpart of rewards) of the state-action pairs in the learning process of TRPO. When the GAIL was proposed in 2017, it was then taken as state-of-the-art on policy reinforcement learning method, TRPO constraints the deviation of the updated policy from the original policy according to their Kullback–Leibler divergence [23]. However later two versions of PPO methods were proposed to first use variable penalty parameters to adjust the KL threshold, and they later used the clipped objective function [8] to reduce the overall computational complexity. We assume the GAIL base on PPO would lead to a more effective result, so we later verify it through our experiment. To explore an effective and socially aware model structure, we set the procedure into three main steps: First, explore a more reliable basic GAIL model, and propose the spatial feature-sensitive version by using LSTM [24], finally extend to the final socially-aware version by aggregate the relative positions of neighborhoods. The content below will introduce this in detail.

3.3. Generative adversarial imitation learning for navigation trajectory

In order to mimic an expert human's trajectory, we decide to implement the classic GAIL algorithm, however, we are curious about which policy optimization would suit such a problem best, so we conduct a comparison experiment the results show that GAIL-PPO has a better convergence speed and accuracy.

Figure 1 demonstrates the overall GAIL structure in the combination of two major parts, the generator and discriminator. The GAIL algorithm adopts the idea of the Generative Adversarial Network (GAN), that is, a generator is trained to generate the corresponding action based on a certain distribution of data to deceive another discriminator trained at the same time. As far as possible, the discriminator cannot distinguish the real trajectory from the generated trajectory; The function of the

discriminator is to distinguish which are true trajectories and which are false trajectories generated by the generator. In such a trade-off process, the model parameters of the two parts are constantly optimized. The goal is for the data distribution generated by the generator to be closest to the real data distribution; During this process, we implement both trust region policy optimization and proximal policy optimization and compared each method's effectiveness.

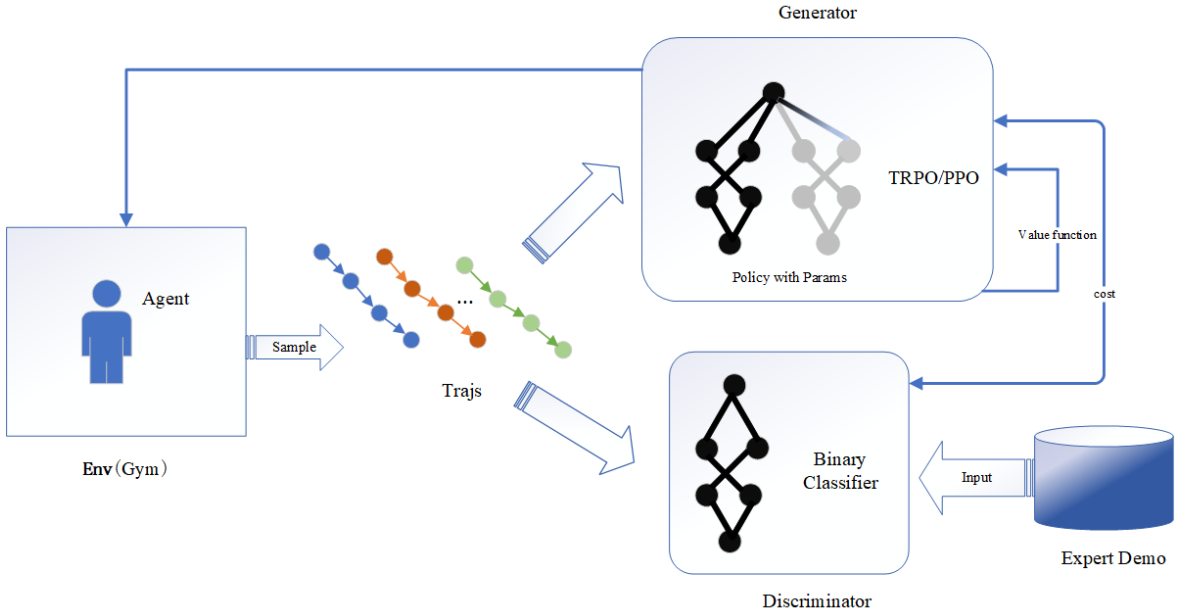

Figure 1. The GAIL with PPO/TRPO as policy optimization.

*3.4. **Temporal enhancement by LSTM***

To better capture the temporal and sequential features, and to help the agent make reasonable actions. We explored the most suitable feature fusion structure and then integrate such LSTM structure into the original GAIL model, in the experiment settings, users could customize a variable to describe the length of previous observations to be considered while making next-step movements.

Assuming the current position of a pedestrian as shown in Figure 2 on the right red arrow parts, we want to know his next position or predict the action he takes from now, traditional GAIL model will only base on previously learned experts distribution to correct the strategically sampled behaviors, but it was found that in the crossing points or interaction parts around some pedestrians, a trained model can appear larger deviation and volatility. Therefore, given this phenomenon, we propose to add LSTM layers, which would be formulated as:

$$\mathbf{h}_{i,t} = LSTM([traj_{i,t-len} : traj_{i,t}], \mathbf{h}_{i,t-1}) \tag{3.2}$$

where $\mathbf{h}_{i,t}$ is the representation of history temporal information based on the current agent $\boldsymbol{i}$'s position at time $\boldsymbol{t}$, and $[traj_{i,t-len} : traj_{i,t}]$ consists of a certain length $\boldsymbol{len}$ of the movement records for extracting the external dynamic features from the frame at $t - len$ to that at the frame t. Then based on the current state, we could aggregate the temporal feature as:

$$\mathbf{aggregated_{input}} = cat(state_t, \mathbf{h}_{i,t}) \tag{3.3}$$

where $state_t$ signifies the direct absolute position tuple at time $\boldsymbol{t}$, we concatenate the two sets of data and input them to an embedding layer for feature fusion. In the previous few steps, LSTM layers could extract certain long temporal characteristics from the hidden layer's output into the model, thus CrowdGAIL can take the history of temporal information and the current state of observation for consideration at the same time. Decision-making based on the feature fusion product is as below:

$$\textbf{action} = MLP(Embedding(\textbf{aggregated}_{input})) \tag{3.4}$$

where the MLP module is a composite of a few Linear () layers with ReLU activation, then as defined in the action space to output the final decision. Making actions based on the former several time steps would be more reasonable and acceptable, and this would improve the forecasting accuracy. And more effective action selection by $feature_{fusion}$ that contains rich temporal dynamics would accelerate the convergence speed.

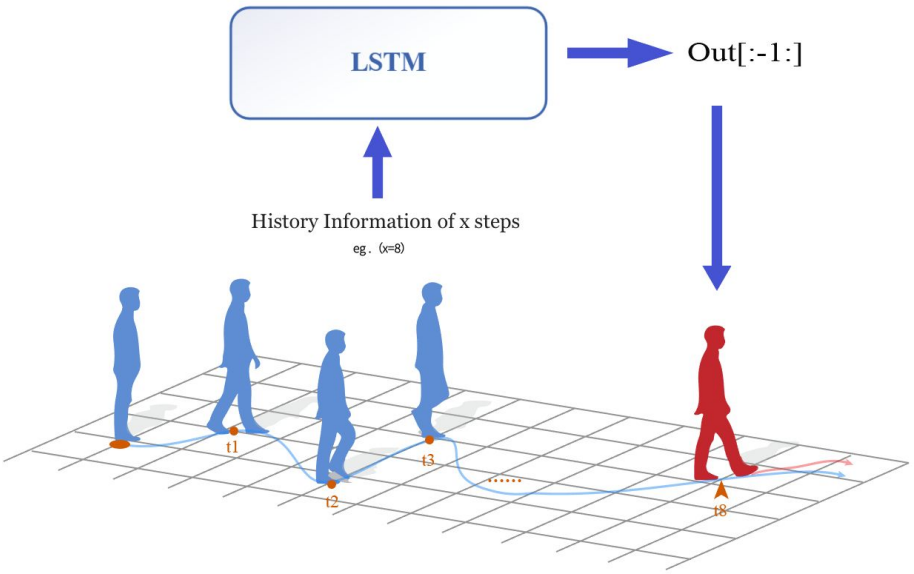

Figure 2. How the LSTM structure stores the spatial info.

*3.5. **Multi agent social manner and crowdGAIL***

In the multi-agent scenario, we borrow the same structure for each agent from the previous proposed LSTM with the sequential demobuffer GAIL, and train the agents in a co-working scene, at the same time, the observation space is expanded from a single view to a global view, which means that the agents could capture others agent's movements to help its own next decision making. In the process, we develop a global information updating mechanism to ensure that the cooperation goes tightly and smoothly. Note that, in this section, the method is still in an early exploring stage, more solid work could be studied in the future.

4. Experiments

In this section, we conduct sufficient experiments in three sub-steps. All the experiments were conducted on a device with Intel(R) Core(TM) i5-4590 CPU@3.30GHz. And for the basic GAIL

implementation, we refer to the GitHub repository: CherryPieSexy for its simplicity and friendly code structure. First, we will demonstrate the comparison between GAIL-TRPO and GAIL-PPO in the same experiment settings. The evaluation metrics are the ADE and FDE [10].

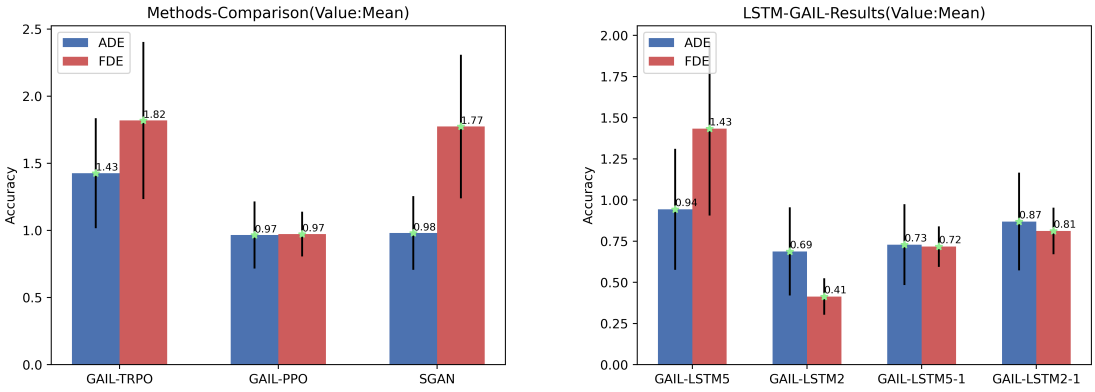

Figure 3. Left:GAIL-TRPO *vs* GAIL-PPO *vs* Social-GAN results; Right:various LSTM structures.

- ADE: *Average displacement error*, (MSE) of all estimated and true points in a trajectory.
- FDE: *Final displacement error*, distance between the predicted and true final destination.

Algorithm 1: CrowdGAIL

Input: Observe length n, expert trajectory $\tau_E \sim \pi_E$, policy and discriminator param θ_0, ω_0

for $i = 0, 1, 2, \ldots$ **do**

Sample ξ_i from DemoBuffer (traj $\tau_i \sim \pi_{\theta_i}$)

Split $Info_{now}$: $\xi_i[t-5:t]$

Calculate $h_{i,t}$ by Eq 3.2

Feature fusion by Eq 3.3

Action output by Eq 3.4

Update discriminator params: $w_i \sim w_{i+1}$

$$\hat{E}_{\tau_i}\left[\nabla_w \log\left(D_w(s,a)\right)\right] + \hat{E}_{\tau_k}\left[\nabla_w \log\left(1 - D_w(s,a)\right)\right] \tag{4.1}$$

Use PPO (clip) to update the policy params: $\theta_i \sim \theta_{i+1}$

$$J_{PPO2}(\theta) \sum_{(\varepsilon_i, a_i)} \min\left[\frac{p_\theta\left(a_t \mid s_t\right)}{p_{\theta'}\left(a_t \mid s_t\right)} A^{\theta'}\left(s_t, a_t\right), \operatorname{clip}\left(\left(\frac{p_\theta\left(a_t \mid s_t\right)}{p_{\theta'}\left(a_t \mid s_t\right)}\right), 1-\epsilon, 1+\epsilon\right) \cdot A^{\theta'}\left(s_t, a_t\right)\right] \tag{4.2}$$

End for

4.1. Study on different policy gradient algorithms & exploration on LSTM structures

In the RL domain, different policy gradient algorithms would have an uneven influence on specific tasks. Developed from REINFORCE, the original GAIL was studied on TRPO by then, however,

with the prosperous and excellent work, PPO showed up a few years ago and has been performing exceedingly in lots of application scenes. We first introduce the PPO-based GAIL to our problem and conducted a fair comparison between the two methods. As shown in Figure 3, on the left hand, the results demonstrate that GAIL-PPO, outperforms GAIL-TRPO due to its effective policy optimization algorithm, and it is also even better on the FDE aspect when compared with SGAN (one of the state-of-the-art methods), the comparison was conducted on 10 cases of testing scenarios, while it was trained on another 10 scenarios. And the actions were taken from the 3 predicted results' average values. The following settings will maintain the same if no special statement. To be more statistical, the error bar was applied to the figure in black vertical line representing the std of results, and the numerical value printed on the bar represents the statistical mean value. It is noticeable that the method of GAIL-PPO not only achieves the highest accuracy on average values but also a more compact range from lowest to highest bound on std. On the right hand, it shows the different history of info fusion structures leading to various results, which in the end, we take the second structure as described in Algorithm 1.

*4.2. **Ablation study on CrowdGAIL and their performances***

Since the proposed method consists of several sub-modules, for a better understanding of their contributions to the preciseness, we conducted an ablation analysis. Figure 4 shows the ablation studies results, please notice that here we haven't integrated social awareness into the model, so we only name it MemoryGAIL for now, the final CrowdGAIL product's performance could be found at the end of Section 4. Here in the below figures, MemoryGAIL represents the structure of **GAIL-PPO-Padding-LSTM**, We could witness the ADE and FDE increasing when cutting off the LSTM (Memory) part, Padding (Sequential Demo Buffer/SDBuffer) part respectively, which verifies the necessity of each module. In general, the GAIL-PPO-Padding version fails to compete with MemoryGAIL in term of both ADE and FDE because the memory module is taken away, leading to the temporal consequence not being well captured, and proving the importance of timing modeling in the agent trajectory projection scenario. When wiping off the DemoBuffer from GAIL-PPO-Padding, leaving a sole GAIL-PPO, both two metrics of evaluation increased (worse), because the data using efficiency is greatly impaired. The fair results comparison is conducted on mean values, std ranges for statistical rationality.

Another metric we adopt is Success Rate (SR), it is a by-product that when the agent is able to mimic human's decision behaviors, they tend to successfully reach the goal. When defining the 'reaching', we describe it as a situation when the final distances between the agents' stopping point and the goals are less than the agents' radius. Since Social-GAN is only trying to predict a certain length of the future trajectory, it eventually failed at reaching the goals. Our results as presented in Table 1.

The collision rate was not compared because we studied some related work, e.g., the CADRL algorithm's motivation is to avoid collision by specially designing a value network to query a collision-free velocity vector, LSTM-RL method mainly intends to better solve the collision rate by introducing LSTM structure and SARL tries to improve the time efficiency while keeping a low collision rate. The above methods heavily rely on designed reward functions to execute any behavior features. Whereas, our CrowdGAIL's purpose is to mimic human's inner policy (intentions) which are not to directly avoid the collision, but to understand trajectory level distributions, like moving in a group way, detouring, etc.

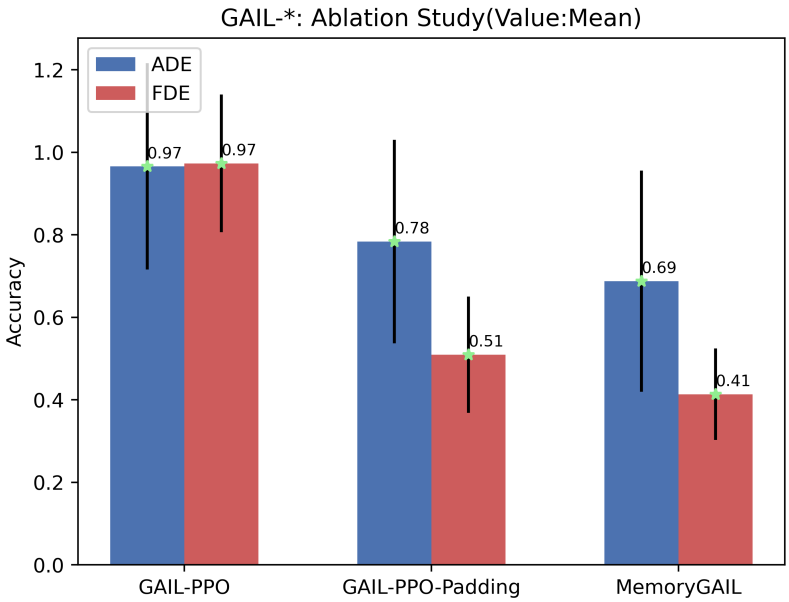

Figure 4. Ablation results of each component of CrowdGAIL.

According to the experimental process, we conduct a case investigation on different methods by applying them to the same scene and then observe the results. As shown in Figure 5, GAIL-TRPO fails to converge on reaching the destination and easy to get deviated. While the second sub-figure of GAIL-PPO slightly remedies such deviation but is still not promising enough. By comparison, the shown Social-GAN's result also suffers severe deviation problems, destination of which is highlighted by Yellow, as it mainly to a failure of long-term predictions. In the end, our model CrowdGAIL shows the most outstanding results both on ADE and FDE.

Table 1. Success rate across different methods.

Methods	Success rate
GAIL-PPO-Padding	0.55
MemoryGAIL	0.82
GAIL-PPO	0.48
GAIL-TRPO	0.11
Social-GAN	0

*4.3. **Initial exploration on social awareness of CrowdGAIL***

By here, we've largely improved the agent's performance by introducing the temporal module and designing a sequential demobuffer. However, our settings are only explored in a single agent. To approximate real life and inspect our method's extensibility, we provided a multi-agent setting. Figure 6 is a demonstration how the social manner and safety distance would influence the decisions. It consists of two main parts, up and down, signifying without/with social awareness, we could monitor the upper movements, and the agents collide with each other. At the same time, the same thing never

happens in the lower half. And the lower agents seem to cultivate a sense of group action by the left bottom two agents because they share a similar direction.

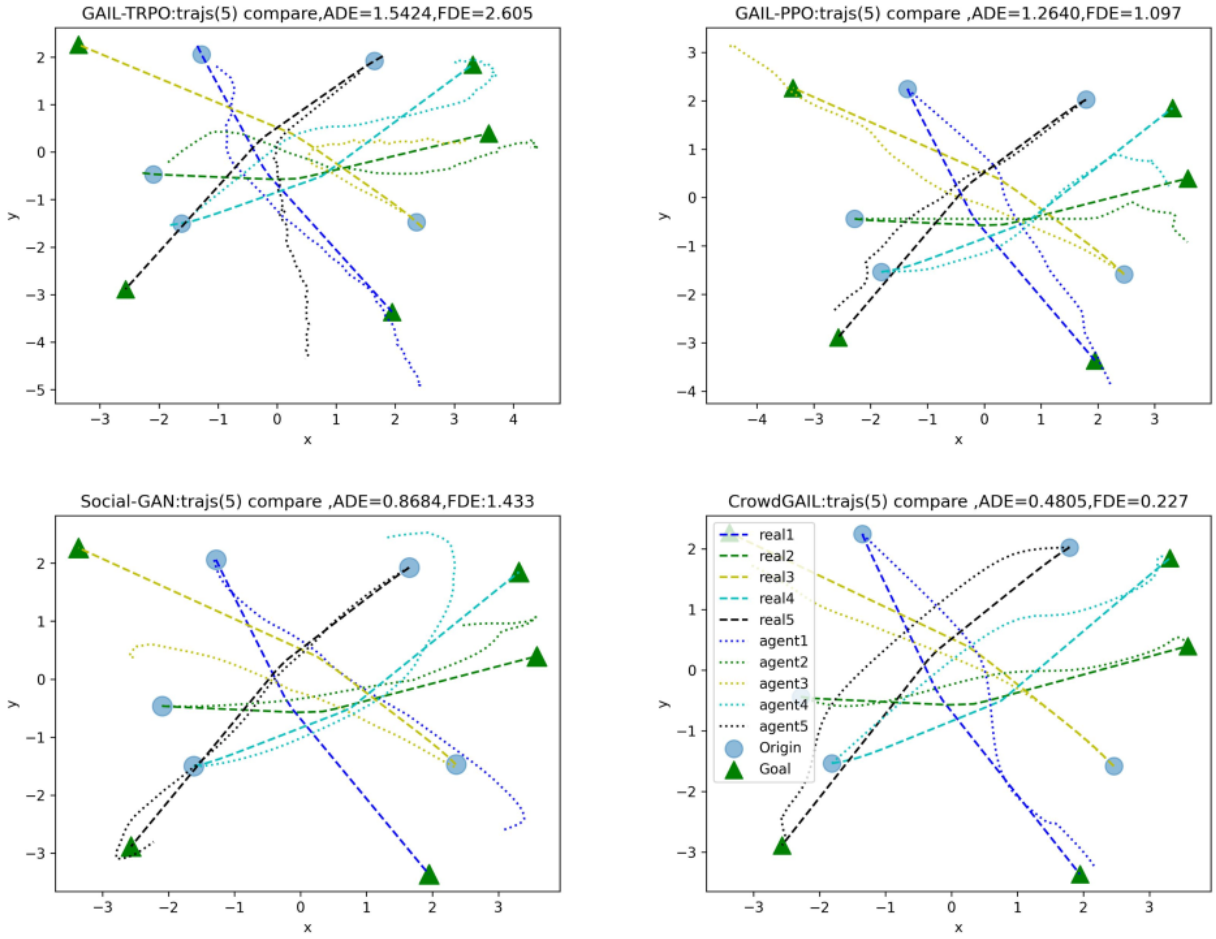

Figure 5. Different methods results comparison: In each sub-figure, a light blue solid circle signifies the starting points of the agents, green triangle signifies the expert traj's real goal. The solid and dashed line means expert trajectory and agent's trajectory. Note that for simplicity, all sub-figures share the same legends shown in the last sub-figure.

5. Conclusions

This paper attempts to address the spatial-temporal aware problem in agent navigation scenarios and realized a method free of the reward function compared to most RL methods. We consider a good behavior policy means that it acts like real human beings when navigating to its goals, and, with more difficulty, there exist social interactions while navigating. The paper developed the CrowdGAIL in three steps, first we analyze different policy optimization approaches, and then base on the GAIL-PPO, we integrate LSTM into the model, after much effort, we developed the most suitable structure

to help the temporal feature fusion, meanwhile, we designed a sequential demo buffer to help out the data's usage efficiency. We analyzed the proposed methods across several baselines, like GAIL-TRPO, GAIL-PPO, Social-GAN, etc. In the end, collision avoidance [25] and social comfort distance were considered, and by the result, we could monitor agents' cooperation and social awareness obviously. However, the social feature was only an attempt in this work, and future deeper study is expected.

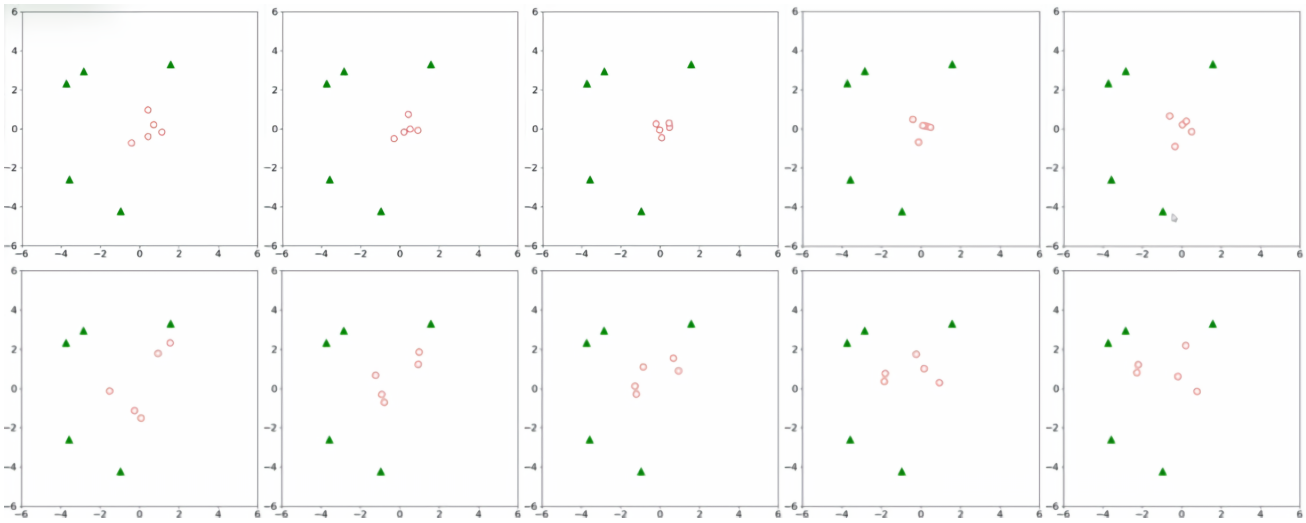

Figure 6. CrowdGAIL with social aware *vs* no aware in a multi-agent scene: Green triangle signifies the destinations of the navigation, the red circle signifies each agent in the scene, from left to right, the upper (no-aware) and lower (with aware) part consists of 5 frames respectively, displaying on the horizontal axis. For an example video, please refer to: https://youtu.be/CSerPna3O9E.

Acknowledgments

Sincerely thank the reviewers for the valuable comments. Thanks to the trust and grant, this work has benefited from funding from the National Science Foundation: IIS-2153311.

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. D. Helbing, P. Molnar, Social force model for pedestrian dynamics, *Phys. Rev. E*, **51** (1995). https://doi.org/10.1103/PhysRevE.51.4282
2. F. Zanlungo, T. Ikeda, T. Kanda, Social force model with explicit collision prediction, *Europhys. Lett.*, **93** (2011). https://doi.org/10.1209/0295-5075/93/68005

3. R. Zhou, Y. Cui, Y. Wang, J. Jiang, A modified social force model with different categories of pedestrians for subway station evacuation, *Tunnelling Underground Space Technol.*, **110** (2021), 103837. https://doi.org/10.1016/j.tust.2021.103837
4. S. Pellegrini, A. Ess, K. Schindler, L. Gool, You'll never walk alone: modeling social behavior for multi-target tracking, in *IEEE 12th international conference on computer vision*, (2009), 261–268. https://doi.org/10.1109/ICCV.2009.5459260
5. T. Fan, X. Cheng, J. Pan, P. Long, W. Liu, R. Yang, et al., Getting robots unfrozen and unlost in dense pedestrian crowds, *IEEE Rob. Autom. Lett.*, **4** (2019), 1178–1185. https://doi.org/10.1109/LRA.2019.2891491
6. A. Ravankar, A. A. Ravankar, Y. Kobayashi, Y. Hoshino, C. Peng, Path smoothing techniques in robot navigation: State-of-the-Art, current and future challenges, *Sensors*, **18** (2018), 3170. https://doi.org/10.3390/s18093170
7. J. Schulman, S. Levine, P. Abbeel, M. Jordan, P. Moritz, Trust region policy optimization, *arXiv preprint*, (2015), arXiv:1502.05477. https://doi.org/10.48550/arXiv.1502.05477
8. J. Schulman, F. Wolski, P. Dhariwal, A. Radford, O. Klimov, Proximal policy optimization algorithms, *arXiv preprint*, (2017), arXiv:1707.06347. https://doi.org/10.48550/arXiv.1707.06347
9. A. Mohamed, K. Qian, M. Elhoseiny, C. Claudel, Social-stgcnn: a social spatio-temporal graph convolutional neural network for human trajectory prediction, in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, (2020), 14424–14432. https://doi.org/10.1109/CVPR42600.2020.01443
10. A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, F. Li, S. Savarese, Social LSTM: human trajectory prediction in crowded spaces, in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2016), 961–971. https://doi.org/10.1109/CVPR.2016.110
11. H. Song, D. Luan, W. Ding, M. Wang, Q. Chen, Learning to predict vehicle trajectories with model-based planning, *arXiv preprint*, (2022), arXiv:2103.04027. https://doi.org/10.48550/arXiv.2103.04027
12. Y. Yuan, X. Weng, Y. Ou, K. Kitani, Agentformer: agent-aware transformers for socio-temporal multi-agent forecasting, in *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, (2021), 9813–9823. https://doi.org/10.1109/ICCV48922.2021.00967
13. A. Hussein, M. M. Gaber, E. Elyan, C. Jayne, Imitation learning: a survey of learning methods, *ACM Comput. Surv.*, **50** (2017), 1–35. https://doi.org/10.1145/3054912
14. A. Aggarwal, M. Mittal, G. Battineni, Generative adversarial network: an overview of theory and applications, *Int. J. Inf. Manage. Data Insights*, **1** (2021), 100004. https://doi.org/10.1016/j.jjimei.2020.100004
15. A. Kuefler, J. Morton, T. Wheeler, M. Kochenderfer, Imitating driver behavior with generative adversarial networks, in *2017 IEEE Intelligent Vehicles Symposium (IV)*, (2017), 204–211. https://doi.org/10.1109/IVS.2017.7995721
16. Y. Mao, F. Gao, Q. Zhang, Z. Yang, An AUV target-tracking method combining imitation learning and deep reinforcement learning, *J. Mar. Sci. Eng.*, **10** (2022), 383. https://doi.org/10.3390/jmse10030383

17. S. Samsani, M. Muhammad, Socially compliant robot navigation in crowded environment by human behavior resemblance using deep reinforcement learning, *IEEE Rob. Autom. Lett.*, **6** (2021), 5223–5230. https://doi.org/10.1109/LRA.2021.3071954

18. C. Chen, Y. Liu, S. Kreiss, A. Alahi, Crowd-robot interaction: crowd-aware robot navigation with attention-based deep reinforcement learning, in *2019 International Conference on Robotics and Automation (ICRA)*, (2019), 6015–6022. https://doi.org/10.1109/ICRA.2019.8794134

19. K. Guo, D. Wang, T. Fan, J. Pan, VR-ORCA: variable responsibility optimal reciprocal collision avoidance, *IEEE Rob. Autom. Lett.*, **6** (2021), 4520–4527. https://doi.org/10.1109/LRA.2021.3067851

20. J. Ho, S. Ermon, Generative adversarial imitation learning, *arXiv preprint*, (2016), arXiv:1606.03476. https://doi.org/10.48550/arXiv.1606.03476

21. L. Mero, D. Yi, M. Dianati, A. Mouzakitis, A survey on imitation learning techniques for end-to-end autonomous vehicles, *IEEE Trans. Intell. Transp. Syst.*, **23** (2022), 14128–14147. https://doi.org/10.1109/TITS.2022.3144867

22. S. Arora, P. Doshi, A survey of inverse reinforcement learning: challenges, methods and progress, *Artif. Intell.*, **297** (2021), 103500. https://doi.org/10.1016/j.artint.2021.103500

23. T. Kim, J. Oh, N. Kim, S. Cho, S. Yun, Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation, *arXiv preprint*, (2021), arXiv:2105.08919. https://doi.org/10.48550/arXiv.2105.08919

24. S. Hochreiter, J. Schmidhuber, Long short-term memory, *Neural comput.*, **9** (1997), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

25. D. Fox, W. Burgard, S. Thrun, The dynamic window approach to collision avoidance, *IEEE Rob. Autom. Mag. IEEE*, **4** (1997), 23–33. https://doi.org/10.1109/100.580977

AIMS Press