

Multi-Robot Cooperative Socially-Aware Navigation Using Multi-Agent Reinforcement Learning

Weizheng Wang¹, Le Mao², Ruiqi Wang¹, and Byung-Cheol Min¹

Abstract—In public spaces shared with humans, ensuring multi-robot systems navigate without collisions while respecting social norms is challenging, particularly with limited communication. Although current robot social navigation techniques leverage advances in reinforcement learning and deep learning, they frequently overlook robot dynamics in simulations, leading to a simulation-to-reality gap. In this paper, we bridge this gap by presenting a new multi-robot social navigation environment crafted using Dec-POSMRD and multi-agent reinforcement learning. Furthermore, we introduce SAMARL: a novel benchmark for cooperative multi-robot social navigation. SAMARL employs a unique spatial-temporal transformer combined with multi-agent reinforcement learning. This approach effectively captures the complex interactions between robots and humans, thus promoting cooperative tendencies in multi-robot systems. Our extensive experiments reveal that SAMARL outperforms existing baseline and ablation models in our designed environment. Demo videos for this work can be found at: <https://sites.google.com/view/samarl>

I. INTRODUCTION

Recent advancements in robotics and artificial intelligence have paved the way for diverse applications such as multi-robot delivery systems [1], collaborative exploration [2], and notably, cooperative navigation in crowded settings [3]. Despite significant research into single-robot social navigation via both reinforcement learning (RL) [4]–[6] and computation theory [7], the multi-robot domain remains ripe for investigation. A predominant challenge lies in defining cooperative robot behaviors while ensuring socially aware pedestrian interactions [8].

Central to this study is the multi-robot cooperative socially-aware navigation (MR-SAN) task (illustrated in Fig. 1). This involves charting feasible, collision-free paths for robots in human-dense environments while respecting social norms. While real-time communication systems facilitate centralized planning for cooperative collision avoidance [9], such systems are not always viable due to potential sensor or computation delays. Therefore, we advocate for a non-communication multi-robot system that operates within the MR-SAN paradigm, where each robot makes decisions based on local observations, accounting for both human-robot interaction (HRI) and robot-robot interaction (RRI).

This paper is based on research supported by the National Science Foundation (NSF) under Grant No. IIS-1846221. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

¹SMART Laboratory, Department of Computer and Information Technology, Purdue University, West Lafayette, IN, USA. [wang5716, wang5357, minb]@purdue.edu. ²College of Mechanical and Electrical Engineering, Beijing University of Chemical Technology, Beijing, China. robotics.lemao@gmail.com.

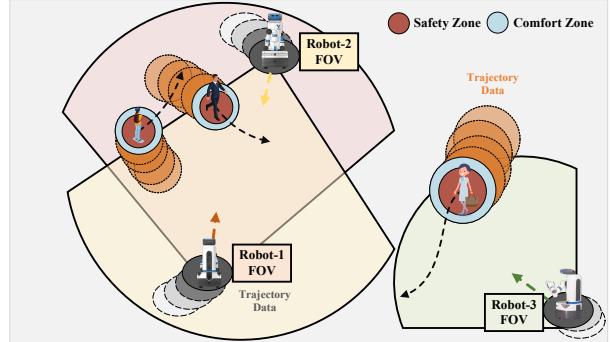


Fig. 1: An illustration of multi-robot cooperative socially-aware navigation: social robots are engaging in cooperative navigation while maintaining a safe social distance from pedestrians.

Existing works in both multi-robot and single-robot social navigation [3], [6], [10], despite their innovations, primarily focus on velocity-driven actions, overlooking the significance of robotic kinematics and dynamics. This omission can introduce a disconnect between simulations and real-world scenarios [11]. Our paper uniquely addresses robot dynamics through non-uniform linear motion constraints and models the environment using a decentralized partially observable semi-Markov decision process (Dec-POSMRD) [12], facilitating more realistic robotic operations.

Decoupled methods [13], [14] often separate HRI inference and path planning, leading to challenges like the freezing robot [15] or reciprocal dance phenomena [16] in complex settings. However, contemporary learning-based approaches [5], [6], [17] have integrated these aspects, demonstrating success in single-robot scenarios. Addressing multi-robot navigation requires managing intricate interactions and representing cooperative paradigms, tasks that traditional single-agent RL algorithms struggle with.

To address these challenges, we present a multi-robot social navigation simulator, leveraging Dec-POSMRD and employing spatio-temporal graph (ST-graph) constructions to capture latent HRI and RRI features across both spatial and temporal dimensions. This data is processed using a hybrid spatial-temporal transformer inspired by [17]. In order to enable effective multi-robot cooperation, we adapt a multi-agent reinforcement learning (MARL) algorithm to train joint policies and facilitate cooperative navigation behaviors.

The main contributions of this paper are: (1) Addressing multi-robot social navigation through a Dec-POSMRD with a focus on realistic dynamics. Our method, called SAMARL, harnesses MARL [18] to guide social robots through human-shared spaces; (2) SAMARL's deployment of a transformer-based HRI and RRI representation network derived from

[17], to infer and align latent spatial and temporal dependencies among robots and pedestrians, facilitating social norm adherence and cooperative behaviors; (3) Demonstrating SAMARL’s superior performance in both simulated and real-world contexts, outpacing current state-of-the-art (SOTA) algorithms.

II. RELATED WORKS

Due to the NP-hard complexity of Bayesian multistep probabilistic inference [19], decoupled methods have faced challenges in the long-term forecasting of crowd movements’ uncertainties. In single-robot scenarios, some approaches explicitly embed constraints into planning strategies, such as using game theory [20] or topology [7]. Alternately, implicitly coupled HRI inference and planning align more closely with human-like thinking paradigm. For instance, in [21], a ST-graph is employed to capture the spatial-temporal dependencies of environmental dynamics, and [22] evaluates cooperation between humans and robots via preference distribution. Additionally, [5] leverages human intelligence and supervision into social navigation using a preference learning framework. More recently, [17] constructs an ST-graph based on attention correlations among spatial-temporal dimensions using a hybrid spatial-temporal transformer to enhance the effectiveness of human preference supervision. While the aforementioned approaches address the single robot socially-aware navigation task effectively, they cannot be directly applied to MR-SAN environments.

In multi-robot settings, [10] models pair-wise HRI for social navigation, which can extend to a multi-agent system. However, simply copying pair-wise correlation to the MR-SAN task leads to the overlook of fully and intrinsic features. For instance, robots need to maintain social distance with pedestrians under HRI and to cooperate with other robots based on RRI embedding. Despite being inspired by the impressive performance of Proximal Policy Optimization (PPO) [23] in various benchmarks, some works [24], [25] adapt single-agent RL algorithm PPO into multi-robot navigation tasks. But as observed in [18], [26], [27], where MARL can effectively study and exhibit cooperation or competition correlations among agents compared to single-agent RL. Generally, recent MARL algorithms train agents to achieve team objectives with the centralized training and decentralized execution (CTDE) paradigm [28]. In this paradigm, each agent has individual actors representing personal policies, while a centralized critic is designed to capture group-wise cooperative embeddings.

More recently, some promising works have introduced a trend that combines RL and advanced neural networks to improve action generation and feature representation. For instance, [29]–[32] have developed transformer networks that abstract high-order distributions of semantic information into a comprehensive range of RL topics, drawing from the success of transformer networks in sequential modeling and the capability of RL in exploration. In this paper, we introduce a hybrid spatial-temporal transformer to capture long-term HRI dependencies, which is trained within a MARL training

framework for multi-robot cooperative motion planning.

III. PRELIMINARY

A. Kinematic & Dynamic Configuration

Many existing learning-based approaches [3], [10], [17], [33], [34] assume that robotic velocities are used as actions directly, denoted as $\mathbf{a} = [v_x, v_y]$, with the robots having infinite acceleration. However, this assumption leads to idealized robotic dynamics characterized by uniform linear motion, which can introduce discrepancies between real physical robots and simulations due to abrupt velocity changes at each time step. To address this issue, we represent robotic actions using acceleration and orientation, denoted as $\mathbf{a}^t = [a_x^t, a_y^t, \theta^t]$, assuming that robots can adjust their velocity orientation in-place at each time step. In reality, physical robots often exhibit non-uniform linear motion. To better capture real-world scenarios, we define a more comprehensive and natural kinematic configuration to model the long-term non-uniform linear motion of robots as follows:

$$\begin{aligned} \mathbf{v}^{t+1} &= \mathbf{v}^0 + \int_0^{t+1} \vec{a}^t dt = \mathbf{v}^t + \vec{a}^t \Delta t \\ \mathbf{p}^{t+1} &= \mathbf{p}^0 + \int_0^{t+1} \mathbf{v}^t dt = \mathbf{p}^t + \mathbf{v}^t \Delta t + \frac{1}{2} \vec{a}^t \Delta t^2 \end{aligned} \quad (1)$$

where Δt is the timestep of the environment, \vec{a}^t is a vector quantity of acceleration at the t -th timestep, and \mathbf{p} denotes the location.

Moreover, we extend Eq. 1 to accommodate scenarios in which certain physical robots are designed to rotate in place, for example, those equipped with Mecanum wheels [35]. In such cases, the current velocity orientation θ becomes a part of action, enabling robots to perform in-place rotation as follows:

$$\begin{aligned} v_x^{t+1} &= \|\mathbf{v}^t\|_2 \cdot \cos \theta + a_x^t \Delta t \\ v_y^{t+1} &= \|\mathbf{v}^t\|_2 \cdot \sin \theta + a_y^t \Delta t \end{aligned} \quad (2)$$

where $\|\cdot\|_2$ represents the L^2 -norm function.

Kinematic constraints of the environments are also introduced to delineate the limitations of simulations in relation to physical robots. These constraints encompass acceleration limits, rotation restrictions, and specified turning radius conditions as follows:

$$\begin{aligned} \|\vec{a}\|_2 &\leq \vec{a}_{\max} \quad ; \quad \left| \lim_{t \rightarrow t^+} \theta^t - \lim_{t \rightarrow t^-} \theta^t \right| \leq \Delta\theta_{\max} \\ \Delta t \cdot \|\mathbf{v}^{t+1}\|_2 &> \left| \lim_{t \rightarrow t^-} \theta^{t+1} - \lim_{t \rightarrow t^+} \theta^t \right| \end{aligned} \quad (3)$$

where \vec{a}_{\max} is the parameterized maximum acceleration of robots, accounting for robotic physical conditions, which is set herein as 5.0 m/s^2 . Also, $\Delta\theta_{\max}$ imposes a limit on the angular range of in-place rotation that robots can perform in one timestep (assuming $\Delta\theta_{\max} = \frac{\pi}{12}$ in the paper). The last equation refers to a minimum turning radius of 1.0 m .

Lastly, in the environmental configuration, we assume that pedestrians have their own policies rather than being treated as simple dynamic obstacles. To maximize the performance of algorithm, we consider scenarios where robots remain unobserved by humans. This takes into account the fact that certain pedestrians, such as elderly individuals or individuals

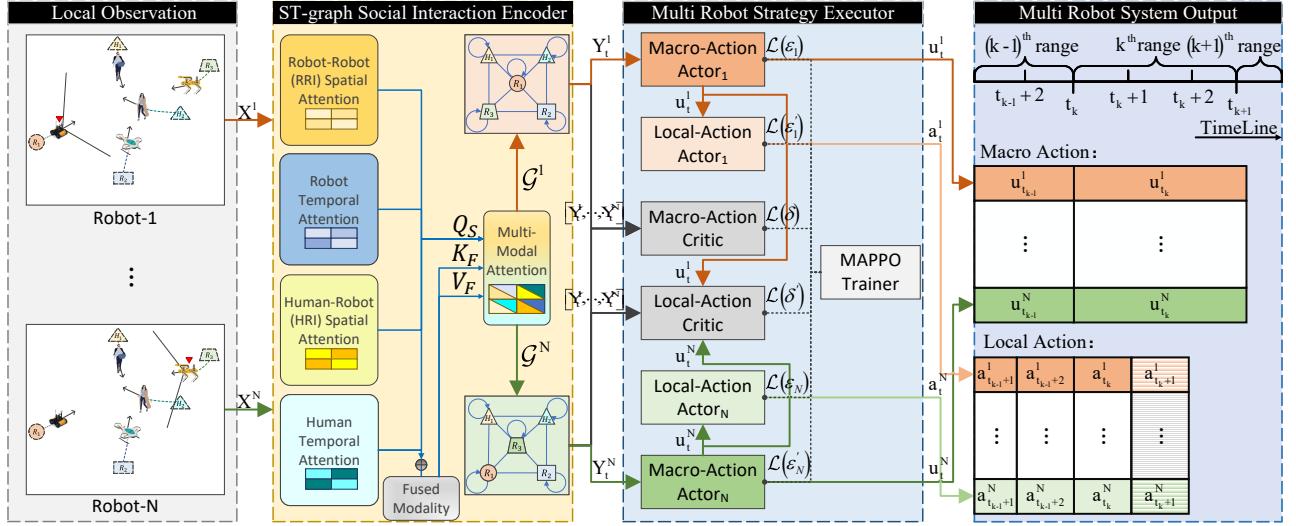


Fig. 2: SAMARL Architecture: First, each robot feeds its individual local observations into the hybrid spatial-temporal transformer-based ST-graph social interaction encoder to create spatial-temporal state representations of HRI and RRI states. Then, the robot leverages environmental dynamics features to perform multi-robot cooperative navigation policies and adhere to social norm, using the MAPPO trainer and the social norm reward function within the multi robot strategy executor block. Finally, the generated macro-action (MA) and local-action (LA) guide the robots in the environment.

engaged in phone calls, may not promptly notice the presence of surrounding robots in real-world conditions.

B. Markov Decision Process Formulation

Taking inspiration from [12], [36], we define the MR-SAN task as a Dec-POSMDP. To accommodate the evolving landscape of social robotic operations, we employ a binary action execution strategy, as outlined in [37]. This strategy consists of macro-action (MA) and local-action (LA). Initially, a MA (\hat{u}) is generated from the multi-agent joint policy $\hat{\pi}$ at each decision-making timestep, typically associated with a global goal or waypoint. Subsequently, a set of LA (\hat{a}) is automatically derived based on the aforementioned MA.

Firstly, the tuple $\langle \mathcal{S}, \mathcal{U}, \mathcal{A}, \Omega, \mathcal{O}, \mathcal{P}, \mathcal{R}, \mathbf{R}, \mathcal{C}, \mathcal{S}_0, \gamma, N \rangle$ is utilized to address MR-SAN as a Dec-POSMDP. $\hat{s}_t = [s_t^{r(1)}, s_t^{o(1)}, \dots, s_t^{r(N)}, s_t^{o(N)}] \in \mathcal{S}^N$ presents that the t -th timestep's joint state belongs to joint state space. The t -th timestep's joint state incorporates all robots' self-states $s_t^{r(\cdot)} = [s_t^{r(\cdot)pu}, s_t^{r(\cdot)pr}]$, which include individual public and privacy information, and their observation state $s_t^{o(\cdot)} = [s_t^{r(\cdot)pu}, \dots, s_t^{h(\cdot)pu}, \dots] \in \Omega$. This observation state can encompass other robots' or pedestrians' observable states with respect to the field of view (FOV) configuration and observable probability \mathcal{O} .

For each agent (robot/human), the individual state includes an observed state and an unobserved state as $s_t = [s_t^{pu}, s_t^{pr}]$. $s_t^{pu} = [p_x, p_y, v_x, v_y, \rho]$ is the agent's public state that can be observed by others. It includes the current position, velocity, and physical radius of the agent. $s_t^{pr} = [g_x, g_y, v_{pref}, \theta]$ represents the agent's privacy state, which is not accessible to others. This state incorporates the agent's personal goal position, preferred speed [10], and heading angle.

\mathcal{U} denotes the joint MA space, and \mathcal{A} represents the joint LA space. At each decision-making timestep $t_k; k \in [0, K]$, every robot's MA $u_{t_k}^i \in \mathcal{U}$ is generated by its policy π^i using the equation $u_{t_k}^i \sim \pi^i(H_{t_k}^i)$, and then all MAs make up a joint MA $\hat{u}_{t_k} \in \mathcal{U}^N$. During each time range

$t \in [t_k, t_{k+1})$, where a total of T timesteps are involved from timestep t_k to timestep t_{k+1} , each robot implements a set of LAs $[a_t^i, \dots, a_{t+T}^i] \sim u_{t_k}^i(H_t^i) \subseteq \mathcal{A}^T$ to interact with the environment. These LAs are obtained based on the currently activated MA to form a joint LA $\hat{a}_t \in \mathcal{A}^N$. H_t^i is the individual history data buffer from timestep t_0 to timestep t .

Ω represents the joint observation space, $\mathcal{O}^i(s^{o(i)} | (s, a)^i)$ denotes the observation probability of the i -th agent, and \mathcal{P} is the state transition probability. $\hat{\mathcal{R}} : \mathcal{S}^N \times \mathcal{U}^N \mapsto \mathbb{R}^N$ is the joint MA reward function with the definition as $\hat{\mathcal{R}}(\hat{s}, \hat{u}) = \mathbb{E}[\sum_{t=0}^T \gamma^t \hat{\mathbf{R}}(\hat{s}_t, \hat{a}_t) | \hat{a}_t \sim \hat{u}(H_t)]$. It consists of the total individual MA rewards \mathcal{R}^i . On the other hand, $\hat{\mathbf{R}} : \mathcal{S}^N \times \mathcal{A}^N \mapsto \mathbb{R}^N$ denotes the joint LA reward function, aggregating all agents' LA rewards \mathbf{R}^i . \mathcal{S}_0 is the initial distribution, N is the number of social robots, and $\gamma \in [0, 1]$ is a discounted factor. Refer to [12] for more definitions.

In addition, the system conditional function \mathcal{C} imposes constraints on the environment and multi-agent system through various termination conditions $\mathcal{C}^i(s_t^i, a_t^i)$. For instance, if a robot reaches its target, it will apply the brakes. Moreover, if the episode exceeds the maximum episode time T or an unfortunate collision occurs between a human-robot or robot-robot, the current episode of the environment will be terminated immediately.

In each episode, the joint initial state of the robots adheres to the initial distribution as $\hat{s}_{t_0} \in \hat{\mathcal{S}}_0$. Initially, each robot's initial MA is obtained as $u_{t_0}^i = \pi^i(H_{t_0}^i)$. Then, the robots update their joint MA at each decision-making moment $t_k; k \in [0, K]$. Meanwhile, the robots interact directly with the environment using LAs $[a_t^i, \dots, a_{t+T}^i] \sim u_{t_k}^i(H_t^i) \subseteq \mathcal{A}^T$, which are unfolded based on their individual current MA during each time range $t \in [t_k, t_{k+1})$. Subsequently, the robots collect joint reward feedback to transition to the next timestep joint state, utilizing the environmental transition probability \mathcal{P} . This process is either terminated or completed based on the responses of the conditional function.

C. Multi Robot Socially-aware Navigation Task Statement

The objective \mathcal{J} of multi robot socially-aware navigation tasks, where N social robots navigate alongside M pedestrians in an open space, can be defined as follows:

$$\begin{aligned} \mathcal{J} &= \arg \min_{\{\tau \in \mathcal{T}, i \in N\}} \sum_{i=1}^N c_i^t(\tau_i) + c_i^s(\tau_i) \\ &= \arg \max_{\hat{\pi}} \mathbb{E} \left[\sum_{k=0}^K \gamma^{t_k - t_0} \hat{\mathcal{R}}(\hat{s}_{t_k}, \hat{u}_{t_k}) \mid (\hat{\pi}, \hat{s}_0) \right] \\ \text{s.t. } & \forall i, j \in [\text{robot}_1, \text{robot}_N], \forall h \in [\text{human}_1, \text{human}_M] \\ & dis_{i,j} > \rho_i + \rho_j, i \neq j; \quad dis_{i,h} > \rho_i + \rho_h; \quad dis_{i,g} < \rho_i \end{aligned} \quad (4)$$

where τ is one of the robotic paths that belong to the clear path region \mathcal{T} , and $c^t(\cdot)$, and $c^s(\cdot)$ are functions representing navigating time or social compliance cost. We relax the optimization of robots' cost functions as a multi-agent expectation function to address Dec-POSMDP problem. The distances between the i -th and j -th robots or the i -th robot and the h -th pedestrian are denoted by $dis_{i,j}$ and $dis_{i,h}$. $dis_{i,g} < \rho_i$ indicates that the i -th robot has reached its target. In summary, we search a joint policy $\hat{\pi} = (\pi^1, \dots, \pi^N)$ for the multi-robot system that can maximize the expectation of the joint MA reward function. This expectation serves as the objective of the Dec-POSMDP problem in the context of MR-SAN. This joint policy generates MAs for the multi-robots while considering the sum of total navigating cost functions for the robots, subject to certain constraints.

IV. METHODOLOGY

A. Social Interaction State Representation

We develop a social interaction state representation framework that infers environmental dynamics using an ST-graph for each robot, as depicted in Fig. 2. This framework is based on a hybrid spatial-temporal transformer [17]. Let each robot's self-state $s_t^{r(\cdot)}$ and observation state $s_t^{o(\cdot)}$ be contained by the robot state $s_t^{(\cdot)}$ as $s_t^{(\cdot)} = [s_t^{r(\cdot)}, s_t^{o(\cdot)}]$, at t -th timestep. The input for the i -th robot's local FOV can be set as $\mathbf{X}^{(i)} = [s_1^{(i)}, \dots, s_t^{(i)}]; i \in [1, N]$. Moreover, we consider the MR-SAN task as a non-communication multi-agent system, where each robot must observe its surrounding environmental information independently. Due to the complexity of navigating in a multi-robot human-filled environment, relying solely on low-order HRI features may not provide sufficient performance and compliance with social norms [38]. Therefore, each robot in SAMARL leverages an individual hybrid spatial-temporal transformer to infer high-order HRI and RRI dependencies in both spatial and temporal dimensions, facilitated by the construction of ST-graphs.

Initially, each robot's local input \mathbf{X} is fed into spatial and temporal transformer networks in parallel to capture spatial and temporal features corresponding to the robot's local FOV. Specifically, due to the different interactive styles and targets between HRI and RRI, observed human and robot states are separately processed by independent spatial-temporal transformer blocks. For instance, SAMARL not only aims to maintain sufficient social distance between robots and pedestrians but also aim to demonstrate cooperative navigation behaviors among robots. Therefore, we divide observed

human states \mathbf{X}_H and observed robot states \mathbf{X}_R to capture different interactions, resulting in RRI spatial feature $\hat{\mathbf{Y}}_{RS}$, robot temporal feature $\hat{\mathbf{Y}}_{RT}$, HRI spatial feature $\hat{\mathbf{Y}}_{HS}$, and human temporal feature $\hat{\mathbf{Y}}_{HT}$.

The spatial transformer refers to [17] that is composed of the positional embedding layer, attention layer, graph convolution layer, and fusion gate with a residual connection. The spatial HRI features, such as correlated importance between a local agent and its neighbors with respect to relative movement intention or the static features (position and velocity) of observed agents in the FOV, can be exhibited using the attention mechanism and the graph convolution framework from spatial transformer. The successes of the vanilla transformer in language modeling, which assesses the relative importance between pairs of words in a sentence [39], is adapted into our framework to determine the importance between each pair of agents in the same timestep.

$$\begin{aligned} \text{Atten}(\mathbf{X}) &= \text{Atten}(\hat{\mathbf{Q}}, \hat{\mathbf{K}}, \hat{\mathbf{V}}) = \text{softmax}\left(\frac{\hat{\mathbf{Q}}(\hat{\mathbf{K}})^\top}{\sqrt{d_h}}\right)\hat{\mathbf{V}} \\ \text{Multi}(\mathbf{X}) &= \text{Multi}(\hat{\mathbf{Q}}, \hat{\mathbf{K}}, \hat{\mathbf{V}}) = f_{fc}(\text{head}_1, \dots, \text{head}_h); \\ \text{head}(\cdot) &= \text{Atten}(\cdot) \end{aligned} \quad (5)$$

where $\hat{\mathbf{Q}}, \hat{\mathbf{K}}, \hat{\mathbf{V}}$ are the query matrix, key matrix, and value matrix of the data \mathbf{X} with a dimension d_h . $f_{fc}(\cdot)$ denotes a fully connected layer, and the maximum head number is h .

Similarly, a temporal transformer is also deployed to calculate the relative importance of each individual with respect to its trajectory history along the temporal dimension. This allows for the inference of agents' self-motion properties, as there is a highly relative temporal dependency in movement. The structure of the temporal transformer is similar to the spatial transformer based on [17], with a graph convolution layer $\text{GCN}(\cdot)$ [40].

$$\begin{aligned} \{\hat{\mathbf{Y}}_{RS}, \hat{\mathbf{Y}}_{HS}\} &= \text{Trans}_{\text{Spa}}(\text{Multi}(\cdot), \text{GCN}(\cdot) \mid \{\mathbf{X}_H, \mathbf{X}_R\}) \\ \{\hat{\mathbf{Y}}_{RT}, \hat{\mathbf{Y}}_{HT}\} &= \text{Trans}_{\text{Tem}}(\text{Multi}(\cdot) \mid \{\mathbf{X}_H, \mathbf{X}_R\}) \end{aligned} \quad (6)$$

Subsequently, a multi-modal transformer form [41], [42] is used to align heterogeneous spatial-temporal features and construct an ST-graph. Initially, the spatial and temporal features are concatenated as a fused feature. Then, the multi-modal transformer queries $\hat{\mathbf{Q}}_S$ each single modality using keys $\hat{\mathbf{K}}_F$ and value $\hat{\mathbf{V}}_F$ from the fused modality, using a multi-head cross-modal attention layer. Finally, a vanilla transformer [39] is deployed to incorporate cross-modality features into an overall state representation \mathbf{Y} as follows:

$$\begin{aligned} \text{CMAAtten}(\mathbf{X}^m) &= \text{Multi}(\hat{\mathbf{Q}}_S^m, \hat{\mathbf{K}}_F, \hat{\mathbf{V}}_F) \\ \mathbf{Y}_t^i &= \text{Trans}_{\text{Mul}}(\text{CMAAtten}(\hat{\mathbf{Y}}_t^i \mid \{\text{RS}, \text{HS}, \text{RT}, \text{HT}\})) \end{aligned} \quad (7)$$

Lastly, the i -th robot ST-graph \mathcal{G}^i is constructed to exhibit latent HRI and RRI with the graphical parameter p_{star}^i . This graph is utilized to generate robot actions and values by the policy π, π' and the value function \mathbf{V}, \mathbf{V}' .

$$\mathbf{Y}_t^i = \mathcal{G}_t^i(\mathbf{X}_t^i; p_{star}^i) \quad (8)$$

Algorithm 1: SAMARL

```

1 Initialize parameters  $(\varepsilon, \varepsilon', \delta, \delta', p_{star}^i, \hat{p}_{star})$ ;
2 while  $step \leq step_{max}$  do
3   set data buffer  $\mathcal{D} = \{\}$ ;
4   for  $i = 1$  to  $batch\_size$  do
5     Reset the environment;
6     Create  $N$  empty caches  $C = [[], \dots, []]$ ;
7     for  $t_k; (k = 0$  to  $K)$  do
8       for all agents  $i = 1$  to  $N$  do
9         if agent  $i$  updates MA  $u_{t_k}^i$  in
10        decision-making timestep  $t_k$  :
11           $\mathcal{G}_{t_k}^i = \pi(o_{t_k}^i; \varepsilon, p_{star}^i, H_{t_k}^i)$ ;
12           $\vartheta_{t_k}^i = \mathbf{V}(\hat{s}_{t_k}; \delta, \hat{p}_{star}, H_{t_k}^i)$ ;
13           $C^i + = [s_{t_k-1}^i, o_{t_k-1}^i, u_{t_k-1}^i, p_{star}^i,$ 
14           $\hat{p}_{star}, H_{t_k}^i, \mathcal{R}_{t_k}^i, s_{t_k}^i, o_{t_k}^i]$ ;
15          Update macro action  $u_{t_k}^i \sim \mathcal{G}_{t_k}^i$ ;
16        end
17        Execute  $a_t^i \sim \pi'(o_{t_k}^i, u_{t_k}^i; \varepsilon', p_{star}^i, H_{t_k}^i)$ ;
18         $\vartheta'_t = \mathbf{V}'(\hat{s}_t, \hat{u}_{t_k}; \delta', \hat{p}_{star}, \hat{H}_t)$ ;
19      end
20      Compute reward and insert data into  $\mathcal{D}$ ;
21    end
22    Update  $(\varepsilon, \varepsilon', \delta, \delta', p_{star}^i, \hat{p}_{star})$  on MAPPO loss;
23 end

```

B. Multi-Agent Reinforcement Learning

We employ one of the current SOTA MARL benchmarks, multi-agent proximal policy optimization (MAPPO) [18], to address the Dec-POSMDP of the MR-SAN task, following the CTDE [28]. During the training procedure, the algorithm utilizes information from all agents to study group-wise cooperative and competitive behaviors in the multi-robot system. However, during execution, each robot only uses individual local observations.

In SAMARL (Algorithm 1), each robot possesses both a decentralized macro-action actor (MA-actor) and local-action actor (LA-actor). The MA-actor produces a global action $u_{t_k} = [\hat{g}_{t_k}^x, \hat{g}_{t_k}^y]$, i.e., the position of next waypoint, during each decision-making timestep $t_k; k \in [0, K]$ using local observation feature \mathbf{Y} to guide the LA-actor in each decision-making time range $[t_k, t_{k+1}]$. The LA-actor, in turn, generates the LA a_t based on the guidance from the MA-actor and local observation features, as shown in Fig. 2. Accordingly, with this setup, both centralized MA critic and LA critic are designed to evaluate the aforementioned actors. They do so based on joint observation features and MA information, thereby assisting the agents in presenting high-order cooperative actions from a global perspective.

SAMARL utilizes MAPPO [18] to train each robot's actor and critic networks with the objective of demonstrating latent cooperation among the multi-robot system. In this approach, MAPPO extends PPO [23] to multi-agent environments through the use of Generalized Advantage Estimation (GAE) [43] and other tricks. The actor drives each robot by generating mean and standard deviation vectors of a multivariate Gaussian distribution from local observations.

The critic evaluates the value of global states to reduce variance. Finally, the parameters for actors ε , critics δ , and the ST-graph p_{star} are updated using the following loss functions $\mathcal{L}(\varepsilon)$ and $\mathcal{L}(\delta)$ as shown in Algorithm 1.

$$\begin{aligned} \mathcal{L}(\varepsilon) &= \sum_{i=1}^N \mathbb{E}_{o \sim \Omega, a \sim \mathcal{A}} [\min\left(\frac{\pi_\varepsilon(a^i|o^i)}{\pi_{\varepsilon_{old}}(a^i|o^i)} \hat{\mathbf{A}}^i, \text{clip}\left(\frac{\pi_\varepsilon(a^i|o^i)}{\pi_{\varepsilon_{old}}(a^i|o^i)}, 1 \pm \epsilon\right) \hat{\mathbf{A}}^i\right) + \kappa \hat{\mathbf{E}}^i] \\ \mathcal{L}(\delta) &= \sum_{i=1}^N \mathbb{E}_{s \sim \mathcal{S}} [\max((\mathbf{V}_\delta(s^i) - \mathbf{R}^i)^2, (\text{clip}((\mathbf{V}_\delta(s^i), \\ &\mathbf{V}_{\delta_{old}}(s^i) - \epsilon', \mathbf{V}_{\delta_{old}}(s^i) + \epsilon') - \mathbf{R}^i)^2)] \end{aligned} \quad (9)$$

where $\hat{\mathbf{A}}$ is the advantage function, which is computed by GAE [43], and $\hat{\mathbf{E}}$ is the policy entropy with an entropy coefficient hyperparameter κ .

The joint reward function $\hat{\mathbf{R}}$ is calculated by each individual reward function $\mathbf{R}(s_t^i, a_t^i)$, and the individual reward function $\mathbf{R}(s_t^i, a_t^i)$ is defined as follows:

$$\mathbf{R}(s_t^i, a_t^i) = \begin{cases} 5, & \text{if } \forall r \in [1, N] \text{ } dis_{r,g} < \rho_r \\ 10, & \text{else } dis_{i,g} < \rho_i \\ -20, & \text{else } s_t \in \mathcal{C}_{\text{collision}}(s_t^i, a_t^i) \\ \max\left(\frac{-1}{dis_{i,h}}, -5\right), & \text{else } dis_{i,h} \leq 0.45 \text{ [44]} \\ 2(dis_{i,g}^{t-1} - dis_{i,g}^t), & \text{otherwise} \end{cases} \quad (10)$$

V. EXPERIMENTS AND RESULTS

A. Simulation Environment

We have created a multi-robot cooperative socially-aware navigation environment, incorporating kinematic and dynamic configurations as described in Section III. Initially, we model the MR-SAN process as a Dec-POSMDP, and then address the Dec-POSMDP using a MARL algorithm with hybrid spatial-temporal transformer. In this environment, each robot's FOV is valued in a range of $(0^\circ, 360^\circ]$, allowing robot to observe states of other robots and pedestrians within FOV area. Humans in the environment follow personal ORCA policies or intents. These human agents are randomly generated along a circle with radius of 5 m. The initial positions of the multi-robots and their respective targets are also determined based on the initial distribution \mathcal{S}_0 .

1) *Baselines and Ablation Study:* We have deployed some SOTA MR-SAN algorithms as baselines for evaluation. Among these, ORCA [45] is viewed as a representative conventional method, while CADRL [10] serves as an example of RL-based algorithms. Moreover, we have introduced the PPO algorithm [23] into the hybrid spatial temporal transformer as an ablation model, additionally, we also have replaced the transformer with SRNN [10]. These ablation models are referred to as SAMARL-PPO and SAMARL-SRNN, respectively, and share the same training parameters.

2) *Training Details:* All the aforementioned algorithms are trained using the same set of environmental hyperparameters. However, the network parameters are configured according to their original papers. Specifically, SAMARL, SAMARL-SRNN and SAMARL-PPO are trained over a total

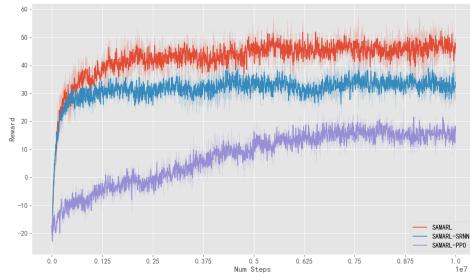


Fig. 3: Learning curves of SAMARL and other two ablation models.

of 1×10^7 timesteps, with a learning rate of 5×10^{-4} for both the actor and critic networks. Other main parameters include PPO-epoch: 5, mini-batch size: 1, gain: 0.01, clip: 0.2, entropy coefficient: 0.01.

3) *Evaluation*: We utilize a joint success rate as our evaluation metric, ensuring that each robot can reach its individual target. Additionally, we modify the single-robot social navigation comprehensive evaluation metric F_{SC} from [17], considering both path quality and social acceptability factors, to create the MR-SAN social score \hat{F}_{SC} :

$$\hat{F}_{SC} = \omega_1 F_{SC}^1 + \dots + \omega_N F_{SC}^N \quad (11)$$

where $\sum_{i=1}^N (\omega_i) = 1$; $\omega_i \in [0, 1]$ is a set of weighted factors.

We conducted experiments with 3 robots and 10 humans, each with a FOV-360°, in 500 random cases. Moreover, we varied the parameters of robot FOV degree, human number, and robot number for more tests, as shown in Table I.

TABLE I: Simulation Experiment Results

| Methods | Success Rate | | | MR-SAN Social Score | | |
|---------------|-----------------|-----------|-----------|---------------------|-----------|-----------|
| | FOV&Human&Robot | | | FOV&Human&Robot | | |
| | 90° | 90° | 360° | 90° | 90° | 360° |
| | 10&3 | 10&5 | 10&5 | 10&3 | 10&5 | 10&5 |
| ORCA [45] | 25 | 21 | 23 | 13 | 8 | 10 |
| CADRL [10] | 55 | 54 | 63 | 37 | 31 | 33 |
| SAMARL-PPO | 72 | 71 | 72 | 68 | 65 | 67 |
| SAMARL-SRNN | 81 | 82 | 87 | 82 | 75 | 79 |
| SAMARL | 93 | 89 | 95 | 94 | 85 | 91 |

4) *Results*: As shown in Table I, the conventional method ORCA exhibits very low performance in terms of both success rate and social score across most environmental configurations. This indicates that ORCA is ill-suited for challenging environments due to its short-sighted one-step lookahead operation. While CADRL performs better than ORCA, it still faces challenges in representing system interactions comprehensively and relies on a limited single-agent learning algorithm (Deep V-learning). The results of CADRL demonstrate that easily repeatable pair-wise interactions cannot accurately model complex HRI and RRI.

As shown in Table I and Fig. 3, the deployments of new configurations generate smoother and reasonable paths than previous works for more realistic scenarios. And SAMARL demonstrates an outstanding performance compared to the other two ablation models, SAMARL-PPO and SAMARL-SRNN, in terms of evaluation metrics such as success rate, social score, and learning effectiveness in the curve. Particularly, in Fig. 4, the first row includes the trajectory visualizations from SAMARL-PPO and SAMARL in the same test case. SAMARL-PPO resulted in an unfortunate collision between a robot and a pedestrian because single agent RL algorithms cannot reasonably capture the mapping

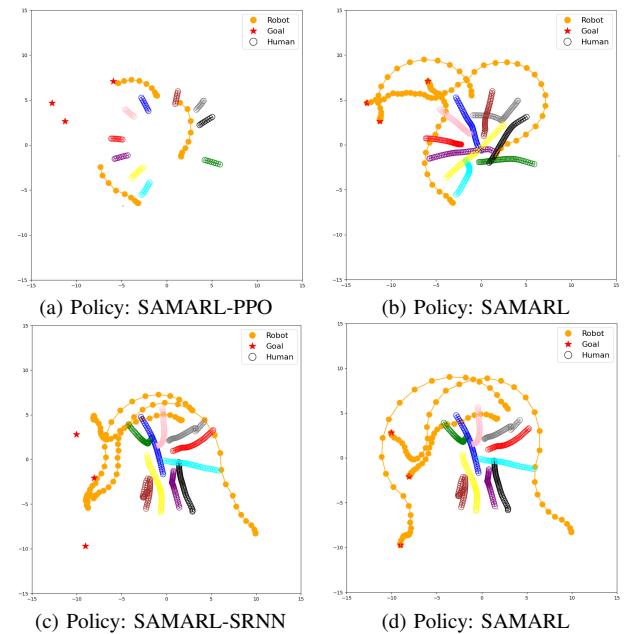


Fig. 4: Comparison Trajectories Visualization: the trajectories visualization of ablation models and SAMARL that are tested by the same test case.

of each agent's local observations to optimized actions. Contrary to SAMARL-PPO, the SAMARL's training Algorithm 1 models each agent's observation-action mapping individually, resulting in better navigation and cooperation behaviors. The second row in Fig. 4 exhibits strategic trends from SAMARL-SRNN and SAMARL, where the joint paths are too close to both humans and robots in SAMARL-SRNN compared to SAMARL. Despite both SAMARL-SRNN and SAMARL utilizing the same MARL method, the ability of social interaction representation still affects performance.

B. Real-world Experiment

We conducted a real-world experiment using two mobile robots, implementing a YOLO [46] & DeepSORT [47] velocity predictor with an objective tracking framework [48] and a 3D localization distance estimator based on [49] as the robot perception system. This system updated observed human or robot states via a Kinect sensor. we involved 5 human participants, each walking towards individual targets while interacting with two social robots in an open space environment of 16m * 16m. The method used to generate human initial positions and targets was the same as in the simulation. As a result, our proposed method SAMARL demonstrated promising performance with an 85.7% success rate in total 14 times real-world test cases¹. Demo videos of our experiments are available on our website.

VI. CONCLUSION

We introduce SAMARL, a benchmark for cooperative social navigation with multi-robots using MARL and transformer networks. It addresses multi-robot socially-aware navigation, incorporating HRI and RRI interactions to achieve cooperative social navigation in complex environments. Our results from simulations and real-world tests affirm its effectiveness, advancing multi-robot navigation.

¹This physical robot experiment was reviewed and approved by the BUCT Institutional Review Board (IRB).

REFERENCES

[1] A. Camisa, A. Testa, and G. Notarstefano, “Multi-robot pickup and delivery via distributed resource allocation,” *IEEE Transactions on Robotics*, vol. 39, no. 2, pp. 1106–1118, 2022.

[2] C. Nieto-Granda, J. G. Rogers III, and H. I. Christensen, “Coordination strategies for multi-robot exploration and mapping,” *The International Journal of Robotics Research*, vol. 33, no. 4, pp. 519–533, 2014.

[3] T. Fan, P. Long, W. Liu, and J. Pan, “Distributed multi-robot collision avoidance via deep reinforcement learning for navigation in complex scenarios,” *The International Journal of Robotics Research*, vol. 39, no. 7, pp. 856–892, 2020.

[4] H. Kretzschmar, M. Spies, C. Sprunk, and W. Burgard, “Socially compliant mobile robot navigation via inverse reinforcement learning,” *The International Journal of Robotics Research*, vol. 35, no. 11, pp. 1289–1307, 2016.

[5] R. Wang, W. Wang, and B.-C. Min, “Feedback-efficient active preference learning for socially aware robot navigation,” in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11336–11343.

[6] S. Liu, P. Chang, Z. Huang, N. Chakraborty, K. Hong, W. Liang, D. L. McPherson, J. Geng, and K. Driggs-Campbell, “Intention aware robot crowd navigation with attention-based interaction graph,” in *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2023, pp. 12015–12021.

[7] C. I. Mavrogiannis and R. A. Knepper, “Multi-agent path topology in support of socially competent navigation planning,” *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 338–356, 2019.

[8] C. Mavrogiannis, F. Baldini, A. Wang, D. Zhao, P. Trautman, A. Steinfeld, and J. Oh, “Core challenges of social robot navigation: A survey,” *ACM Transactions on Human-Robot Interaction*, vol. 12, no. 3, pp. 1–39, 2023.

[9] A. Gasparri, L. Sabattini, and G. Ulivi, “Bounded control law for global connectivity maintenance in cooperative multirobot systems,” *IEEE Transactions on Robotics*, vol. 33, no. 3, pp. 700–717, 2017.

[10] Y. F. Chen, M. Liu, M. Everett, and J. P. How, “Decentralized non-communicating multiagent collision avoidance with deep reinforcement learning,” in *2017 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2017, pp. 285–292.

[11] A. Kadian, J. Truong, A. Gokaslan, A. Clegg, E. Wijmans, S. Lee, M. Savva, S. Chernova, and D. Batra, “Sim2real predictivity: Does evaluation in simulation predict real-world performance?” *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6670–6677, 2020.

[12] S. Omidshafiei, A.-A. Agha-Mohammadi, C. Amato, S.-Y. Liu, J. P. How, and J. Vian, “Decentralized control of multi-robot partially observable markov decision processes using belief space macro-actions,” *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 231–258, 2017.

[13] C. Cao, P. Trautman, and S. Iba, “Dynamic channel: A planning framework for crowd navigation,” in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 5551–5557.

[14] N. E. Du Toit and J. W. Burdick, “Robot motion planning in dynamic, uncertain environments,” *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 101–115, 2011.

[15] P. Trautman, J. Ma, R. M. Murray, and A. Krause, “Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation,” *The International Journal of Robotics Research*, vol. 34, no. 3, pp. 335–356, 2015.

[16] J. K. Johnson, “The colliding reciprocal dance problem: A mitigation strategy with application to automotive active safety systems,” in *2020 American Control Conference (ACC)*. IEEE, 2020, pp. 1417–1422.

[17] W. Wang, R. Wang, L. Mao, and B.-C. Min, “Navistar: Socially aware robot navigation with hybrid spatio-temporal graph transformer and preference learning,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 11348–11355.

[18] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, “The surprising effectiveness of ppo in cooperative multi-agent games,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 24611–24624, 2022.

[19] G. F. Cooper, “The computational complexity of probabilistic inference using bayesian belief networks,” *Artificial intelligence*, vol. 42, no. 2-3, pp. 393–405, 1990.

[20] A. Turnwald and D. Wollherr, “Human-like motion planning based on game theoretic decision making,” *International Journal of Social Robotics*, vol. 11, pp. 151–170, 2019.

[21] S. Liu, P. Chang, W. Liang, N. Chakraborty, and K. Driggs-Campbell, “Decentralized structural-rnn for robot crowd navigation with deep reinforcement learning,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 3517–3524.

[22] M. Sun, F. Baldini, P. Trautman, and T. Murphey, “Move Beyond Trajectories: Distribution Space Coupling for Crowd Navigation,” in *Proceedings of Robotics: Science and Systems*, Virtual, July 2021.

[23] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, “Proximal policy optimization algorithms,” *arXiv preprint arXiv:1707.06347*, 2017.

[24] R. Han, S. Chen, and Q. Hao, “Cooperative multi-robot navigation in dynamic environment with deep reinforcement learning,” in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 448–454.

[25] P. Long, T. Fan, X. Liao, W. Liu, H. Zhang, and J. Pan, “Towards optimally decentralized multi-robot collision avoidance via deep reinforcement learning,” in *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 2018, pp. 6252–6259.

[26] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, and R. Vicente, “Multiagent cooperation and competition with deep reinforcement learning,” *PloS one*, vol. 12, no. 4, p. e0172395, 2017.

[27] J. G. Kuba, R. Chen, M. Wen, Y. Wen, F. Sun, J. Wang, and Y. Yang, “Trust region policy optimisation in multi-agent reinforcement learning,” in *International Conference on Learning Representations*, 2022. [Online]. Available: <https://openreview.net/forum?id=EcGGFkNTxdJ>

[28] R. Lowe, Y. I. Wu, A. Tamar, J. Harb, O. Pieter Abbeel, and I. Mordatch, “Multi-agent actor-critic for mixed cooperative-competitive environments,” *Advances in neural information processing systems*, vol. 30, 2017.

[29] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, “Decision transformer: Reinforcement learning via sequence modeling,” *Advances in neural information processing systems*, vol. 34, pp. 15084–15097, 2021.

[30] M. Wen, J. Kuba, R. Lin, W. Zhang, Y. Wen, J. Wang, and Y. Yang, “Multi-agent reinforcement learning is a sequence modeling problem,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 16509–16521, 2022.

[31] L. Meng, M. Wen, C. Le, X. Li, D. Xing, W. Zhang, Y. Wen, H. Zhang, J. Wang, Y. Yang *et al.*, “Offline pre-trained multi-agent decision transformer,” *Machine Intelligence Research*, vol. 20, no. 2, pp. 233–248, 2023.

[32] R. Wang, D. Zhao, and B.-C. Min, “Initial task allocation for multi-human multi-robot teams with attention-based deep reinforcement learning,” in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 7915–7922.

[33] C. Chen, S. Hu, P. Nikdel, G. Mori, and M. Savva, “Relational graph learning for crowd navigation,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2020, pp. 10007–10013.

[34] C. Chen, Y. Liu, S. Kreiss, and A. Alahi, “Crowd-robot interaction: Crowd-aware robot navigation with attention-based deep reinforcement learning,” in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6015–6022.

[35] Z. Hendzel *et al.*, “Modelling of dynamics of a wheeled mobile robot with mecanum wheels with the use of lagrange equations of the second kind,” *International Journal of Applied Mechanics and Engineering*, vol. 22, no. 1, pp. 81–99, 2017.

[36] D. S. Bernstein, R. Givan, N. Immerman, and S. Zilberstein, “The complexity of decentralized control of markov decision processes,” *Mathematics of operations research*, vol. 27, no. 4, pp. 819–840, 2002.

[37] A. Loquercio, E. Kaufmann, R. Ranftl, M. Müller, V. Koltun, and D. Scaramuzza, “Learning high-speed flight in the wild,” *Science Robotics*, vol. 6, no. 59, p. eabg5810, 2021.

[38] Y. Liu, Q. Yan, and A. Alahi, “Social nce: Contrastive learning of socially-aware motion representations,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 15118–15129.

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” *Advances in neural information processing systems*, vol. 30, 2017.

[40] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks,” in *International Conference on Learning Representations*, 2017.

- [41] Y.-H. H. Tsai, S. Bai, P. P. Liang, J. Z. Kolter, L.-P. Morency, and R. Salakhutdinov, “Multimodal transformer for unaligned multimodal language sequences,” in *Proceedings of the conference. Association for Computational Linguistics. Meeting*, vol. 2019. NIH Public Access, 2019, p. 6558.
- [42] R. Wang, W. Jo, D. Zhao, W. Wang, A. Gupte, B. Yang, G. Chen, and B.-C. Min, “Husformer: A multi-modal transformer for multi-modal human state recognition,” *IEEE Transactions on Cognitive and Developmental Systems*, 2024.
- [43] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, “High-dimensional continuous control using generalized advantage estimation,” *International Conference on Learning Representations (ICLR)*, 2016.
- [44] J. Rios-Martinez, A. Spalanzani, and C. Laugier, “From proxemics theory to socially-aware navigation: A survey,” *International Journal of Social Robotics*, vol. 7, pp. 137–153, 2015.
- [45] J. Van Den Berg, S. J. Guy, M. Lin, and D. Manocha, “Reciprocal n-body collision avoidance,” in *Robotics Research: The 14th International Symposium ISRR*. Springer, 2011, pp. 3–19.
- [46] “Yolov8.” [Online]. Available: <https://docs.ultralytics.com/>
- [47] N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” in *2017 IEEE international conference on image processing (ICIP)*. IEEE, 2017, pp. 3645–3649.
- [48] A. Pramanik, S. K. Pal, J. Maiti, and P. Mitra, “Granulated rcnn and multi-class deep sort for multi-object detection and tracking,” *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 6, no. 1, pp. 171–181, 2021.
- [49] L. Bertoni, S. Kreiss, and A. Alahi, “Monoloco: Monocular 3d pedestrian localization and uncertainty estimation,” in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 6861–6871.