Autonomous Navigation in Crowded Space Using Multi-sensory Data Fusion

Nourin Siddique Ananna, Mollah Md Saif, Maisha Noor, Ishrat Tasnim Awishi,
Md. Khalilur Rhaman, and Md. Golam Rabiul Alam
Department of Computer Science and Engineering, BRAC University, Dhaka, Bangladesh

***Abstract*—Autonomous navigation in crowded environments remains a significant challenge due to the highly dynamic and unpredictable nature of pedestrian movements. This paper presents a novel approach for socially-compliant crowd navigation by leveraging human pose tracking, trajectory prediction, and obstacle avoidance techniques. We introduce *PoseTrajNet*, an end-to-end autonomous agent navigation pipeline that integrates YOLOv8 for object detection, BlazePose for real-time human pose estimation, and a custom trajectory prediction model drawing on concepts from Social GANs. PoseTrajNet employs pose keypoints as socially-compliant features to anticipate pedestrian trajectories, enabling proactive path planning and dynamic safe radius adjustments for obstacle avoidance. Extensive evaluations on standard datasets demonstrate PoseTrajNet's effectiveness in seamless crowd navigation, outperforming baselines while adhering to social norms.**

***Index Terms*—Autonomous Agents; Human-Aware Motion Planning; Social HRI; Localization; Collision Avoidance; Motion and Path Planning; Data Sets for Robot Learning; Sensor Fusion; Vision-Based Navigation;**

I. Introduction

With the increasing popularity of autonomous driving, the popularity of mobile robot navigation in both indoor and outdoor spaces has increased. However, mobile robot navigation in crowded or dynamic places encounters many challenges- a lack of physical datasets, freezing zone problem in dynamic areas, human safety around robots, and many more. To ensure human-robot safety, research work [1]–[5] has worked on prediction of robot trajectory by incorporating a social compliance feature. Their works have proved that for robot autonomous navigation, both local agents and the entire context of the scene need to be considered. However, most works are based on simulation or in a controlled environment. Thus, it is important to bring in datasets and models which will not only take the social features into account but also provide a dense crowd scenario. This social feature becomes even more important as robots tend to get into freezing problem [6], when the robot stops abruptly due to lack of planning because so many obstacles are close by. To address these challenges, this research makes the following key contributions:

- We introduce a novel **PoseTrajNet** model, which leverages pose tracking, trajectory prediction, and obstacle avoidance using multisensory fusion data.
- A comprehensive dataset named **BU-Crowd** has been developed, featuring multisensory crowd data collected within the BRAC University premises, ensuring social compliance.
- Development of a cost-effective indoor localization method utilizing grid tiled floor patterns, enabling accurate positioning without dedicated visual markers.
- Design of a dynamic obstacle avoidance algorithm that adapts its safe radius based on predicted future obstacle locations, enhancing navigation in crowded environments.
- Demonstration of the navigation system's effectiveness through real-world experiments in diverse crowded environments.

In this paper, Section I provides an introduction and context for our study. Section II outlines the background and related work that informs our research. Section III details the dataset used in our experiments. Section IV describes the methodology applied in our study in detail. Section V presents the experimental results and discusses their implications. Section VI concludes the paper and outlines future directions.

II. Related Works

To navigate dense crowds Chen [1] used deep reinforcement learning while Alahi [7] used Social GAN to train trajectory prediction models. With dense crowd, robots tend to freeze since they are surrounded by humans [6]. This raises the question of human-robot safety, especially in crowded spaces. Socially aware navigation [3] introduces an LSTM based GAN model with social compliancy features which reduces the freezing problem. However, due to the computational power and the lack of real-world experiment, it makes them dificult to use in the real world. Our study leverages PointPillars, pose tracking, and trajectory prediction to navigate among crowds.

A. PointPillars

PointPillars offers an efficient framework for processing LiDAR data in 3D object detection [8]. By grouping points into pillars and encoding them with neural networks, it enables compatibility with standard 2D convolutional architectures. This approach significantly reduces computational demands while preserving the full information of point clouds through end-to-end learning.

Fig. 1: Dataset collection setup utilizing a sensor suite including a stereo camera, depth camera, INS, wheel encoder, and 360° camera. The left panel shows a grid of four images depicting RGB, depth, and point cloud data captured during collection. The center image displays our data collection robot equipped with the depth camera and INS. The right panel illustrates the processing pipeline, showcasing point cloud data, object detection, and pose estimation results

B. Pose Tracking

We face a unique set of challenges in estimating the pose of multiple people in images. Firstly, the number of people varies in an image that can appear at any position or scale. Secondly, complex spatial interference happens because of interactions between people. It makes the association of parts difficult. And lastly, with the number of people, image runtime complexity tends to grow, which makes real-time performance challenging. Openpose [9], a groundbreaking approach for real-time multi-person 2D pose estimation using Part Affinity Fields. This bottom-up system achieves high accuracy and efficiency by leveraging PAFs to associate body parts with individuals in the image, without relying on global context information. Similarly, in Alphapose [10], a realtime framework for multi-person full body pose estimation and tracking, performed using Symmetric Integral Keypoint Regression (SIKR), Parametric Pose Non-Maximum-Suppression (P-NMS) and Pose Aware Identity Embedding. Their approach follows a top-down frame work where it first detects human bounding boxes and independently estimates the annotated 136 points pose of the whole body within each box. Both Alphapose and Openpose are computationally expensive, which is not ideal for implementation in mobile robots.

C. Trajectory prediction

RobustTP [11] computes the trajectories among dense traffic using a combination of non-linear motion model and a deep learning-based instance segmentation algorithm. Taking the trajectories detected as input, RobustTP uses a custom trajectory prediction model using Social-GAN [4], Covolutional Social-LSTM [12], TraPHic [13] and RNN Encoder-Decoder [14]. Another algorithm, DenseCAvoid [15] integrates Deep Reinforcement Learning (DRL)-based approaches with pedestrian trajectory prediction-based navigational tools. Therefore, the system provides benefits of learning-based methods when it comes to dealing with noisy-sensor data. Both DenseCAvoid and RobustTP are going to be computationally expensive for implementing on a robot in real-time.

III. DATASET

During the investigative phase of this study, a robust custom dataset, **BU-Crowd** was collected to enhance the training process. This dataset comprises images of densely populated areas within BRAC University and encompasses three distinct categories of data. Along with our custom dataset, we have also used two publicly available datasets - SCAND [16] and MuSuHo [17] to also train our models. We have pre-processed the data in SCAND and MuSuHo to be able to use them in our robots. The ETH dataset, utilized for simulation experiments, is both large and diverse, with a pair of cameras mounted on a mobile platform comprising of 12,298 annotated pedestrians across approximately 2,000 frames, making it a robust resource for evaluating pedestrian detection and tracking models.

- **Static dataset:** Static dataset was taken in a static frame of reference using a RGB-D camera. The camera mounted on a tripod was placed in the middle of a dense human crowd. This type of data demonstrates the social behaviours in human movement in crowds. This social feature is used for predicting social compliant human trajectories. To get possible human poses, we trained our dataset with a pose detection model and used that to train human trajectory prediction models.
- **Dynamic dataset taken walking**: A human walked around with the sensors among dense crowds and took RGB-D, inertial data of the environment. This data helped understand how humans move in crowded environments and mimic that feature in our dataset. This gave our data a human-human interaction feature, which we could use to further understand the social compliant features to mimic human-like movement in the crowd.
- **Dynamic dataset taken on our Robots**: The final dataset was taken on a four-wheeler robot in a dense environment. The primary sensors were placed on the

robot and were driven in crowded areas to get various sensor data with robot movement. The primary data taken are RGB-D feed, lateral and reverse camera feed, inertial and odometry data, GNSS data where line of sight of satellite is available, and various robot movement data.

A. Sensor Suit

- **Stereo and Depth Camera:** We have used Intel RealSense Depth Camera D435i for RGB, RGB-D, and Point cloud datasets. The RealSense D435i is equipped with an in-built high-accuracy 6-axis IMU which helps video stabilization post-processing.
- **Inertial Navigation System (INS):** Inertial Navigation System (INS) is used in navigation and motion control systems to determine the position, orientation, and velocity by measuring its acceleration and angular velocity. We are using SBG Systems Ellipse-D, capable of delivering precise heading as well as centimeter level position accuracy in the most challenging GNSS conditions.
- **Odometry:** Odometry data is collected from the encoder of the robot wheels. We also get supplementary odometry data from the INS. This helps track the robot's position relative to a starting point. By continuously updating the position as the robot moves, it can maintain awareness of its location within an environment.

B. Dataset features

BU_Crowd is a sizable dataset, comprising approximately 12 hours, 412 trajectories, and 18 kilometers of navigation data collected by two types of four-wheeled robots in various public spaces. The dataset's multi-modal nature is a key strength, as it provides rich sensory information which will give us a more accurate trajectory prediction for both indoors and outdoors. The dataset covers diverse indoor and outdoor environments, exposing a wide range of social navigation scenarios. Importantly, it includes detailed annotations of social interaction events encountered during navigation, such as navigating against heavy crowds, overtaking pedestrians, and navigating through crowds. This context awareness is particularly valuable, as it reflects the nuanced adaptations humans make in their navigation behaviors based on the situational urgency or time constraints.

C. Dataset Use cases

The multi-modal nature, diverse environments, detailed annotations, and context awareness of our dataset make it a valuable resource for training autonomous robots through techniques like imitation learning, inverse reinforcement learning, or reinforcement learning. By leveraging this rich dataset of human demonstrations, robots can potentially acquire the ability to navigate public spaces seamlessly, adhering to unwritten social norms while safely and efficiently reaching their goals.

TABLE I: Tags, Categories and Counts of BU-Crowd

Tag Group	Tag Category	Count
Traffic	Extremely Dense	67
	Dense	85
	Sparse	170
	Extremely Sparse	105
Spatial Configuration	Blind Corner	45
	Narrow Alleyway	204
	Wide Space	240
Crowd Interaction	Waiting in Crowd	89
	Crossing through a Crowd	107
	Bypassing a Crowd	365

D. Dataset Collection and Analysis

The data collection process utilized three distinct modalities to capture multi-modal social interactions during navigation scenarios. A mobile robot platform was equipped with an integrated sensor suite comprising a RealSense camera, a 360-degree camera, an Inertial Navigation System (INS), and a GNSS antenna. Additionally, a hand-carried sensor rig and static sensor deployments were employed. The robot-mounted and hand-carried sensors interfaced through the Robot Operating System (ROS) framework, enabling data synchronization and storage in *rosbag* files. Pre-processing involved stabilizing video data using gyroflow stabilization, aligning point cloud and depth data with RGB data, and passing inertial/odometry data through an extended Kalman filter for precise localization. 3D pose estimation and object detection outputs were also published as dataset features. We have ensured accurate multi-modal data fusion can be done for comprehensive analysis of captured human-human and human-robot interactions.

IV. METHODOLOGY

Our paper proposes a novel navigation pipeline consisting of two systems as shown in figure 2; **a local path planner** and **a global path planner**. The **local path planner** only concerns itself with its surrounding range of 10m and is used for pose tracking and obstacle avoidance. **The global planner** takes into consideration the entire room or environment and is used for future trajectory prediction and path planning. Our robot takes three types of sensor data; RGB camera feed, odometry, and depth data. The camera feed and depth data are used for object detection and pose tracking, whereas the odometry data is used for localisation and odometry systems. The camera feed helps identify humans with YOLOv8 and get the coordinates of the person with respect to the robot. The pose estimation model identifies 33 landmarks for the human and the estimated pose is stored as keypoints. The estimated keypoints and the coordinates with respect to the robot are fed into the Future Trajectory system in the Global Planner as social compliance features.

In the **global planner**, we use our future trajectory model, **PoseTrajNet**, inspired by the social GAN, to predict the

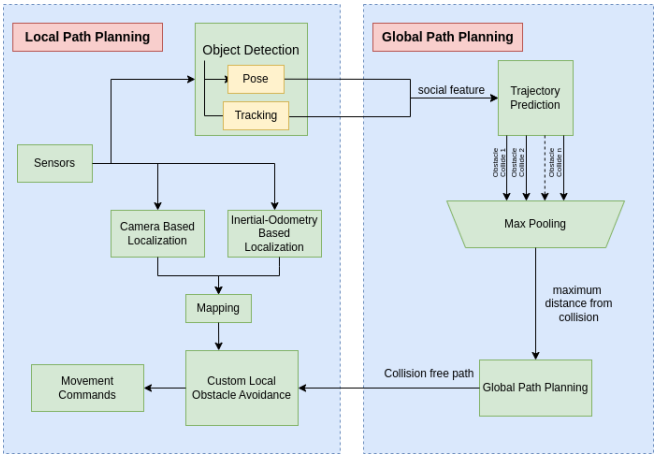


Fig. 2: Proposed Navigation Pipeline for Autonomous System

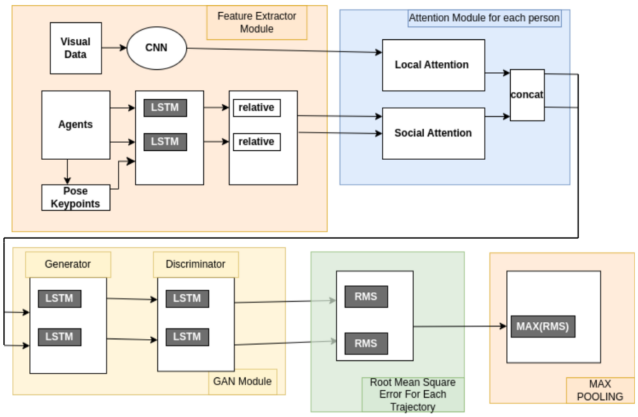


Fig. 3: PoseTrajNet: The proposed Model Architecture for Trajectory Prediction

future trajectory of a person. According to Figure[3], the keypoints and the co-ordinates are fed into PoseTrajNet which can be broken down into the attention module and the GAN module. The attention module inspired by Social-GAN and LSTM modules [18] [19], combines the idea of social attention with local attention that helps the model to look at the bigger picture by highlighting the important information for the LSTM-based GAN module. The GAN module takes the highlighted portion and generates three possible trajectories. We then perform max pooling on them [2] where the trajectory with maximum root mean square (RMS) is chosen, which indicates a maximum distance between robot and the person reducing the chances of a collision. After the trajectory is chosen, it is sent to the path planning system in the **global path planner** and the path generated is sent to the object avoidance system in the **local path planner**. In the object avoidance system, the robot figures out the action to be taken, i.e. move forward, change direction or stop based on the given path and the odometry data from the IMU in that instance. If there is a person, the robot changes the direction to a certain angle and then comes back to its original angle after the path is cleared.

A. Tracking and Pose Estimation

To navigate an autonomous robot through crowds, we must predict human trajectories and avoid obstacles. This is challenging and can lead to the ”freezing zone” problem where the robot fails to predict trajectories effectively [6]. We employ YOLOv8 for real-time object detection, segmentation, pose estimation, and tracking [20]–[22]. For multi-person pose tracking, we use BlazePose [23], a lightweight neural network designed for mobile and web applications. BlazePose combines heatmap prediction with direct keypoint coordinate regression and uses a novel tracker to follow individuals across frames. It identifies 33 body landmarks and handles complex poses by focusing on rigid body parts like faces or hips.

Algorithm 1 PoseTrajNet Trajectory Prediction Algorithm

Require: Data inputs: visual, pose, social_agents
Ensure: Training a LSTM-GAN modules with attention
$F \leftarrow \{\text{visual}, \text{pose}, \text{social\_agents}\}$ ▷ Initialize data inputs
$LSTM_{\text{local}}, LSTM_{\text{social}} \leftarrow$ Initialize LSTM modules
$G, D \leftarrow$ Initialize GAN (Generator, Discriminator)
for $t = 1, \ldots, t_{\text{total}}$ **do**
  $F_{\text{visual}} \leftarrow \text{CNN}(\text{visual})$
  $F_{\text{pose}} \leftarrow LSTM_{\text{local}}(\text{pose})$
  $F_{\text{social}} \leftarrow LSTM_{\text{social}}(\text{social\_agents})$
  $L_{\text{attn}} \leftarrow \text{LocalAttention}(F_{\text{pose}})$
  $S_{\text{attn}} \leftarrow \text{SocialAttention}(F_{\text{social}})$
  for each agent i **do**
    $S_{\text{attn}}[i] \leftarrow \text{Attention}(\text{social\_agents}_i, \{\text{social\_agents}_j\}_{j \neq i})$
  end for
  $F_{\text{combined}} \leftarrow \text{concat}(L_{\text{attn}}, S_{\text{attn}})$
  $T \leftarrow \text{GANModule}(F_{\text{combined}})$
  $RMS_T \leftarrow \text{ComputeError}(T)$
  $T_{\text{max}} \leftarrow \text{MaxPooling}(RMS_T)$
  if $t \bmod t_{\text{eval\_freq}} = 0$ **then**
    Evaluate T_{max}
  end if
  if $t \bmod t_{\text{learn\_freq}} = 0$ **then**
    Train GAN module with batch sample
  end if
end for

B. Robust Indoor Localization

Robust indoor localization is crucial for autonomous navigation in crowded environments. Traditional methods, such as those relying on visual landmarks like AR markers or QR codes, often face challenges due to extensive setup requirements and potential occlusions. To address these limitations, we propose a novel tile-based indoor navigation localization approach that combines visual odometry and inertial sensing. The method implements a 6D state space representation under constant acceleration to track floor tiles with Lucas-Kanade optical flow and to fuse accelerometers and gyroscopes within a Kalman filter. Through median displacement calculation and robust feature tracking, the algorithm allows accurate location even in the case of visual occlusions. Edge detection and Hough transforms enable a metric-scale estimate of tile size without map-building of the environment beforehand. Gyroscope based rotation estimation takes care of sensor integration, and aligns the visual with inertial coor-

TABLE II: Quantitative results including both displacement metrics (ADE/FDE) and the robot stopping performance metric, Time Not Moving (TNM, in seconds). Lower values indicate better performance.

	Baselines					**PoseTrajNet (ours)**				
Dataset	Lin	LSTM	S-LSTM	S-GAN	S-GAN-P	TA	To + Io	To + IA	TAP + I	TAP + IA
ETH	1.33 / 2.94 / 3.20	1.09 / 2.41 / 2.85	1.09 / 2.35 / 2.80	0.81 / 1.52 / 2.45	0.87 / 1.62 / 2.50	0.88 / 1.58 / 2.40	0.84 / 1.62 / 2.35	0.69 / 1.45 / 2.10	0.75 / 1.52 / 2.25	**0.68 / 1.40 / 2.00**
HOTEL	0.39 / 0.72 / 1.50	0.86 / 1.91 / 2.80	0.79 / 1.76 / 2.70	0.72 / 1.61 / 2.60	0.67 / 1.37 / 2.45	0.86 / 1.80 / 2.75	0.82 / 1.77 / 2.70	0.69 / 1.76 / 2.40	0.80 / 1.77 / 2.55	**0.73 / 1.62 / 2.30**
ZARA1	0.62 / 1.21 / 2.10	0.41 / 0.88 / 1.80	0.47 / 1.00 / 1.90	0.34 / 0.69 / 1.65	0.35 / 0.68 / 1.70	0.36 / 0.67 / 1.75	0.32 / 0.67 / 1.65	0.33 / 0.65 / 1.60	0.30 / 0.62 / 1.55	**0.28 / 0.59 / 1.50**
AVG	0.79 / 1.59 / 2.60	0.70 / 1.52 / 2.50	0.72 / 1.54 / 2.47	0.58 / 1.18 / 2.23	0.61 / 1.21 / 2.25	0.70 / 1.35 / 2.30	0.66 / 1.35 / 2.30	0.57 / 1.29 / 2.10	0.62 / 1.30 / 2.13	**0.56 / 1.20 / 2.00**

dinate frame. Extensive experiments conducted substantiate a drastic reduction in drift in comparison with standard visual odometry methods, making it a core model in autonomous systems for indoor navigation.

This tile-based localization can be seamlessly integrated with other sensors and algorithms, such as inertial measurement units (IMUs) and simultaneous localization and mapping (SLAM) techniques, to create a comprehensive and adaptable indoor positioning system. The low-cost and low-maintenance nature of our approach makes it an attractive option for widespread adoption, enabling autonomous robots, drones, and other mobile systems to navigate complex indoor settings with ease.

C. Dynamic Obstacle Avoidance Algorithm

A velocity-obstacle-based approach has been employed to dynamically generate avoidance maneuvers, ensuring safe navigation through crowded environments [24]. Similarly, a reactive control method has been utilized to enable continuous adaptation to pedestrian movements in densely crowded settings [25]. However, these methods are computationally expensive [8], [25]. For this, we propose specialized algorithm proposed that enables robot navigation in environments with dense human populations. It starts with a global path plan and anticipates collisions by predicting future obstacle movements. A trajectory cloud is then created within a calculated safe radius to avoid these obstacles. The algorithm adjusts this radius dynamically, using a PID controller to navigate the robot towards the safe zone's boundary. Tailored for areas with high pedestrian traffic, characterized by unpredictable movements, our approach updates the safe radius based on new obstacle predictions to better adapt to evolving scenarios.

$$R_s(t + \Delta t) = \max(R_{min}, k \cdot \|\mathbf{p}_{new} - \mathbf{p}_{robot}\|) \quad (1)$$

Here, R_s represents the safe radius, R_{min} the minimum allowable radius, k a scaling factor, p_{new} the position of the new obstacle, and p_{robot} the position of the robot. This formula showcases how our algorithm integrates global path planning with real-time obstacle avoidance to maintain both effective navigation and safety. The dynamic adjustment of the safe radius and the ability to re-plan paths allow our model to effectively handle varying obstacle densities and configurations.

V. EXPERIMENTS RESULTS AND EVALUATION

This section explores the details of both simulation experiments and real-world testing for evaluating the PoseTrajNet model. We start with an evaluation on common benchmarks of the trajectory prediction problem, both quantitative metrics and qualitative analysis of future trajectory predictions. We then report on experimental work with a custom built robot in multiple dense real-world environments.

A. Simulation Experiment

We evaluate our proposed method using two prominent datasets: ETH [26] and UCY [27]. We benchmark our model against the baselines on these datasets. We also performed a qualitative analysis to examine the effectiveness of the future trajectory predictions in our models. For baseline comparisons and ablation studies, we utilize two datasets: ETH [26] and UCY [27]. They encompass five distinct scenes: Univ (from UCY), and ETH and Hotel (from ETH). Each scene includes top-view images and 2D coordinates of individuals relative to world coordinates, with static cameras capturing the scenes.

We compare our PoseTrajNet model against several baselines, including Linear regression (Lin), LSTM, Social LSTM (S-LSTM), Social GAN (S-GAN), and Social GAN with Pooling (S-GAN-P). Our model variations include Trajectory Attention (TA), Trajectory-only + Input-only (To + Io), Trajectory-only + Input Attention (To + IA), Trajectory Attention and Pose + Input (TAP + I), and our full model Trajectory Attention and Pose + Input Attention (TAP + IA).

For quantitative evaluation, we use three metrics: Average Displacement Error (ADE) and Final Displacement Error (FDE), both measured in meters, and Time Not Moving (TNM), measured in seconds. ADE is defined as the mean L2 distance between the ground truth and predicted positions over all predicted time steps, while FDE represents the distance between the predicted final destination and the true final destination at the end of the prediction period. Additionally, TNM quantifies the total duration during which the predicted trajectory indicates the agent remains stationary when a stop is expected, offering further insights of the prediction.

Table II presents the results of our experiments, reporting values in the format "ADE / FDE / TNM." Our full model (TAP + IA) consistently outperforms the baselines and other variations across all datasets, with an average ADE/FDE/TNM of 0.56/1.20/2.00, compared to the next best baseline (S-GAN) with 0.58/1.18/2.23.

1) Training Model PoseTrajNet: The PoseTrajNet model features a generator and a discriminator, both trained iteratively using the Adam optimizer, with a mini-batch size of 64 and a learning rate of 0.005 over the course of 100 epochs.

The encoded trajectory representation is input to a long short-term memory (LSTM) network with a 32-dimensional hidden layer in the generator. In contrast, the discriminator utilizes an LSTM with a 64-dimensional hidden layer to handle the encoded trajectories. The generator's decoder employs a single-layer MLP with a 16-dimensional embedding to encode agent positions, which are then processed by an LSTM with a 32-dimensional hidden layer. Our custom dataset, BU-Crowd, was used in training PoseTrajNet by extracting human movement patterns from both the static and dynamic datasets. The static dataset contributed insights into social behaviors in crowd movement, while the dynamic datasets from both human-carried and robot-mounted sensors provided real-world motion patterns essential for learning socially compliant human-like trajectories.

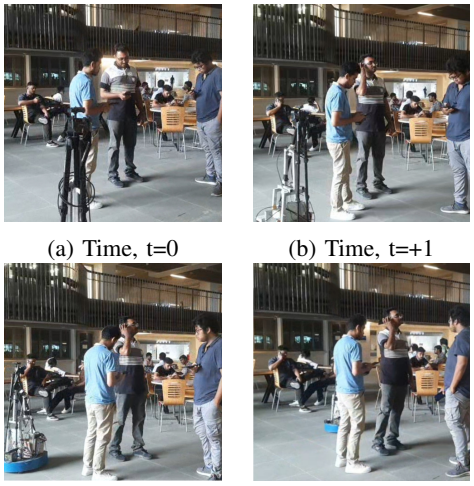
2) Attention Mechanisms: The social attention component calculates attention weights by processing the encoder output and decoder context through several MLP layers with sizes 64, 128, 64, and 1, interspersed with ReLU activations. The output of the final layer is then passed through a Softmax layer. The module accounts for interactions with up to 32 surrounding agents (Nmax), as no scenes in the datasets have more than 32 active agents at any timestep. If fewer agents are present, the remaining slots are filled with a dummy value of 0. The local attention module takes raw VGG features (512 channels), projects them using a convolutional layer, and embeds them using a single MLP to an embedding dimension of 16. It is important to note that the discriminator does not utilize the attention modules or the decoder network.

TABLE III: Test-bed Navigation Performance: Success Rate, Collision Rate, and Completion Time (s). Bold indicates best performance.

Method	Success	Collision	Time
TA	0.43	0.57	13.27
To + Io	0.78	0.22	13.10
To + IA	0.95	0.03	14.48
TAP + I	1.00	0.00	12.83
TAP + IA	**1.00**	**0.00**	**11.28**

B. TEST-BED IMPLEMENTATIONS

To evaluate the performance and robustness of our PoseTrajNet model, we conducted extensive real-world experiments and test-bed implementations using our custom-built robot platform. Our robot, equipped with a suite of sensors including RGB-D cameras, IMU, and odometry, was deployed in various crowded environments such as university campuses, shopping malls, and public spaces. While performing experiments, the robot navigated autonomously, relying on the PoseTrajNet model for pose tracking, obstacle avoidance, future trajectory prediction, and path planning. The model's ability to incorporate socially compliant features, such as pose keypoints, and adapt to real-time changes in the environment was closely monitored and assessed. Throughout the extensive testing, the robot successfully navigated through the crowded environments without any collisions, demonstrating the effectiveness of the obstacle avoidance and path planning components of the PoseTrajNet model. The robot's navigation performance, including success rate, collision avoidance, and social compliance, was evaluated against ground truth data and compared to baseline methods. The results from these real-world experiments demonstrated the effectiveness and practicality of the PoseTrajNet model in enabling safe, efficient, and socially acceptable robot navigation in complex crowded scenarios, validating its potential for real-world applications.



(a) Time, t=0 (b) Time, t=+1

(c) Time, t=+2 (d) Time, t=+3

Fig. 4: Real-world demonstration of the proposed model in crowded environments, showing four sequential time frames of the model's performance

VI. CONCLUSIONS

In our work, we have proposed **PoseTrajNet** a novel trajectory pipeline for socially-compliant autonomous crowd navigation which integrates human pose tracking, trajectory prediction using pose keypoints as social features, and dynamic obstacle avoidance through safe radius adjustments. Key innovations include multi-sensor fusion for robust localization, heading correction algorithms, and the creation of a custom crowded environment dataset.The custom dataset, **BU-Crowd** offers a unique and extensive collection of data captured in densely populated environments marking a significant contribution to the crowd navigation field. The dataset includes detailed annotations of human poses and trajectories, providing critical insights into social interactions and movement patterns in crowds. Extensive evaluations demonstrated PoseTrajNet's ability to seamlessly navigate crowds while adhering to social norms. This paves the way for the deployment of socially aware robots in highly crowded public spaces like the university by addressing challenges in pedestrian prediction, obstacle avoidance, and socially acceptable navigation. Future work will focus on improving the crowded dataset.

REFERENCES

[1] Y. F. Chen, M. Everett, M. Liu, and J. P. How, "Socially aware motion planning with deep reinforcement learning," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 1343–1350.

[2] 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)*, 2019, pp. 6015–6022.

[3] K. Li, M. Shan, K. Narula, S. Worrall, and E. Nebot, "Socially aware crowd navigation with multimodal pedestrian trajectory prediction for autonomous vehicles," in *2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC)*, 2020, pp. 1–8.

[4] A. Gupta, J. Johnson, L. Fei-Fei, S. Savarese, and A. Alahi, "Social gan: Socially acceptable trajectories with generative adversarial networks," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 2255–2264.

[5] T. Fan, X. Cheng, J. Pan, D. Manocha, and R. Yang, "Crowdmove: Autonomous mapless navigation in crowded scenarios," *ArXiv*, vol. abs/1807.07870, 2018. [Online]. Available: https://api.semanticscholar.org/CorpusID:49904993

[6] A. J. Sathyamoorthy, U. Patel, T. Guan, and D. Manocha, "Frozone: Freezing-free, pedestrian-friendly navigation in human crowds," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, pp. 4352–4359, 2020.

[7] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 961–971.

[8] A. H. Lang, S. Vora, H. Caesar, L. Zhou, J. Yang, and O. Beijbom, "Pointpillars: Fast encoders for object detection from point clouds," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 689–12 697.

[9] Z. Cao, G. Hidalgo, T. Simon, S.-E. Wei, and Y. Sheikh, "Openpose: Realtime multi-person 2d pose estimation using part affinity fields," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 172–186, 2021.

[10] H.-S. Fang, J. Li, H. Tang, C. Xu, H. Zhu, Y. Xiu, Y.-L. Li, and C. Lu, "Alphapose: Whole-body regional multi-person pose estimation and tracking in real-time," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 6, pp. 7157–7173, 2023.

[11] R. Chandra, U. Bhattacharya, C. Roncal, A. Bera, and D. Manocha, "Robusttp: End-to-end trajectory prediction for heterogeneous road-agents in dense traffic with noisy sensor inputs," in *Proceedings of the 3rd ACM Computer Science in Cars Symposium*, ser. CSCS '19. New York, NY, USA: Association for Computing Machinery, 2019. [Online]. Available: https://doi.org/10.1145/3359999.3360495

[12] N. Deo and M. M. Trivedi, "Convolutional social pooling for vehicle trajectory prediction," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2018, pp. 1549–15 498.

[13] R. Chandra, U. Bhattacharya, A. Bera, and D. Manocha, "Traphic: Trajectory prediction in dense and heterogeneous traffic using weighted interactions," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 8475–8484.

[14] D. Britz, A. Goldie, M.-T. Luong, and Q. Le, "Massive exploration of neural machine translation architectures," in *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, M. Palmer, R. Hwa, and S. Riedel, Eds. Copenhagen, Denmark: Association for Computational Linguistics, 2017, pp. 1442–1451. [Online]. Available: https://aclanthology.org/D17-1151

[15] A. J. Sathyamoorthy, J. Liang, U. Patel, T. Guan, R. Chandra, and D. Manocha, "Densecavoid: Real-time navigation in dense crowds using anticipatory behaviors," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 11 345–11 352.

[16] H. Karnan, A. Nair, X. Xiao, G. Warnell, S. Pirk, A. Toshev, J. Hart, J. Biswas, and P. Stone, "Socially compliant navigation dataset (scand): A large-scale dataset of demonstrations for social navigation," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 11 807–11 814, 2022.

[17] D. M. Nguyen, M. Nazeri, A. Payandeh, A. Datar, and X. Xiao, "Toward human-like social robot navigation: A large-scale, multi-modal, social human navigation dataset," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 7442–7447.

[18] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 1349–1358.

[19] P. Zhang, W. Ouyang, P. Zhang, J. Xue, and N. Zheng, "Sr-lstm: State refinement for lstm towards pedestrian trajectory prediction," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 12 077–12 086.

[20] A. Bochkovskiy, C.-Y. Wang, and H.-Y. M. Liao, "Yolov4: Optimal speed and accuracy of object detection," *ArXiv*, vol. abs/2004.10934, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:216080778

[21] J. Dai, H. Qi, Y. Xiong, Y. Li, G. Zhang, H. Hu, and Y. Wei, "Deformable convolutional networks," in *2017 IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 764–773.

[22] S. Liu, J. Chi, and C. Wu, "Fcos-lite: An efficient anchor-free network for real-time object detection," in *2021 33rd Chinese Control and Decision Conference (CCDC)*, 2021, pp. 1519–1524.

[23] V. Bazarevsky, I. Grishchenko, K. Raveendran, T. L. Zhu, F. Zhang, and M. Grundmann, "Blazepose: On-device real-time body pose tracking," *ArXiv*, vol. abs/2006.10204, 2020. [Online]. Available: https://api.semanticscholar.org/CorpusID:219793039

[24] P. Fiorini and Z. Shiller, "Motion planning in dynamic environments using velocity obstacles," *The International Journal of Robotics Research*, vol. 17, no. 7, pp. 760–772, 7 1998. [Online]. Available: https://doi.org/10.1177/027836499801700706

[25] D. Paez-Granados, Y. He, D. Gonon, D. Jia, B. Leibe, K. Suzuki, and A. Billard, "Pedestrian-robot interactions on autonomous crowd navigation: Reactive control methods and evaluation metrics," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 149–156.

[26] S. Pellegrini, A. Ess, and L. V. Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," in *European Conference on Computer Vision*, 2010. [Online]. Available: https://api.semanticscholar.org/CorpusID:16060712

[27] A. Lerner, Y. Chrysanthou, and D. Lischinski, "Crowds by example," *Computer Graphics Forum*, vol. 26, no. 3, pp. 655–664, 2007. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/j.1467-8659.2007.01089.x