POLITECNICO DI TORINO

MASTER 's Degree in MECHATHRONICS ENGINEERING

MASTER 's Degree Thesis

Learning Heuristics for Adaptive Path Planning in Social Scenartios

Supervisor

Prof. Marcello CHIABERGE

Candidate

Filippo AISA

OCTOBER 2024

Abstract

Autonomous robots are increasingly adopted in environments shared with humans, where adherence to social norms is essential for seamless integration. Traditional navigation algorithms, while effective in ensuring efficiency and safety, often neglect the social aspects of human-robot interaction, leading to behaviors that can disrupt human activities or cause discomfort.

This thesis presents a novel, unified, learning-based framework for socially aware navigation. Given a map that also encodes people's positions and orientations, a convolutional neural network computes a social cost layer, enabling robots to navigate human environments while respecting social conventions. By integrating social norms into the decision-making process, the proposed approach goes beyond obstacle avoidance to include socially appropriate behavior in dynamic, multi-agent environments.

The framework was developed and validated through simulations at the PIC4SeR center, where social navigation is a major research topic. It demonstrated its effectiveness in various social scenarios, such as navigating crowded spaces, interacting with groups, queueing properly, and keeping right in hallways. The results highlight the framework's potential to enhance human-robot coexistence in public spaces, making autonomous systems more acceptable and effective in real-world applications.

Table of Contents

List of Figures

Chapter 1

Introduction

1.1 Background and Motivation

Autonomous navigation in environments shared with humans is a complex task that extends beyond traditional robotics challenges. It requires robots to not only avoid obstacles but also adhere to social norms that govern human interactions. This area of research, commonly referred to as socially aware navigation, has become increasingly important as robots are expected to operate in public spaces such as shopping malls, airports, and hospitals. The ability of robots to navigate in a socially compliant manner is essential for their acceptance and integration into human environments.

Despite significant advances in autonomous navigation, many existing algorithms focus primarily on efficiency and safety, often neglecting the subtle social cues that humans rely on during navigation. This oversight can result in robotic behaviors that are perceived as unnatural or disruptive by humans, increasing resistance in the widespread adoption of autonomous systems in everyday life.

1.2 Research Problem

The primary challenge in socially aware navigation lies in the development of algorithms that enable robots to move in human environments while respecting social conventions. These conventions include maintaining appropriate interpersonal distances, avoiding disruption of human activities, and predicting human intentions. Existing methods typically address these challenges in a fragmented manner, focusing on individual aspects of social navigation without providing a comprehensive solution.

This thesis aims to address this gap by developing a unified socially aware navigation framework, the aim is to create a navigation system that not only ensures safety and efficiency but also exhibits behavior that is perceived as socially appropriate by humans.

1.3 Thesis Objectives

The main objective of this thesis is to develop a unified framework for socially aware navigation that allows robots to navigate seamlessly in environments shared with humans. Specifically, the objectives of this research are:

1. To review and analyze existing socially aware navigation frameworks and identify their limitations.
2. To develop a unified model that integrates social norms into the robot's decision-making process.
3. To validate the proposed framework through simulation, assessing its effectiveness in various simulated social scenarios.

1.4 Thesis Structure

The remainder of this thesis is organized as follows. Chapter 2 provides a detailed literature review of existing socially aware navigation methods and their limitations. Chapter 3 presents the methodology proposed, detailing the integration of social norms in the planning process. Chapter 4 describes the experimental setup used for validation. Finally, Chapter 5 discusses the results, followed by conclusions and potential directions for future research in Chapter 6.

Chapter 2

State of the Art

2.1 Introduction

Socially aware navigation has become a crucial capability for robots that interact with humans in shared environments. As robots move from structured industrial environments to unstructured public spaces such as malls, hospitals, and airports, the need for socially compliant navigation is critical. This chapter outlines the evolution of socially aware navigation, starting from early approaches and progressing toward state-of-the-art methods in Local Planning and Global Path Planning.

2.2 Early Approaches to Socially Aware Navigation

The earliest socially aware navigation systems treated humans merely as dynamic obstacles. Projects such as RHINO [1] and MINERVA [2], implemented in museum tour guide robots, successfully navigated human environments by avoiding collisions. However, these systems lacked an understanding of social conventions such as maintaining personal space or respecting human interactions.

The introduction of proxemics, which describes how humans maintain personal and social space, marked a major advancement. Sisbot et al. [3] incorporated proxemics into robot navigation, using human position and orientation to dynamically adjust the robot's path. This led to the development of methods such as ROS's social navigation layers, which inflate the robot's cost map around humans to prevent uncomfortable interactions.

Despite these improvements, early systems were primarily reactive and lacked the foresight necessary for more complex social scenarios such as queuing or navigating around groups. This limitation led to the development of more sophisticated techniques in both local and global planning.

2.3 Local Planning

Local planning focuses on creating short-term trajectories based on real-time sensing and a provided global path to follow, and giving velocity commands to execute the computed trajectories.
Local planners have to deal with the circumstances unaccounted for in the generation of the global path, such as unexpected obstacles, or, in the social navigation scenario, humans.
Several approaches exist for the generation of the local trajectory:

2.3.1 Planning-Based Approaches

Planning-based approaches generate trajectories based on a model of the environment and the robot's motion. One commonly used approach is Model Predictive Control (MPC), which generates an optimal trajectory by solving an optimization problem over a finite time horizon. This method accounts for both the robot's dynamics and environmental constraints, but it can also include social constraints if a social model is available, like in the work presented by Eshan et al. [4].

Another popular method is the Dynamic Window Approach (DWA), which generates a set of reachable velocities based on the robot's kinematic constraints and selects the best velocity that brings the robot closer to its goal while avoiding obstacles. Truong and Ngo [5] applied DWA in human environments, ensuring safe navigation through dynamic human crowds. Similarly, Kabtoul et al. [6] introduced dynamic channels to anticipate pedestrian cooperation in crowded environments, further enhancing human-robot interaction.

Elastic Band Approaches generate paths that are dynamically adjusted by simulating an elastic band that stretches and compresses around obstacles. The timed elastic band approach, proposed by Rösman et al. [7], incorporates kinodynamic constraints, making it suitable for dynamic human environments. Khambhaita and Alami [6] employed elastic bands to proactively plan around humans, ensuring safe and socially aware navigation.

2.3.2 Force-Based Approaches

Force-based approaches model human-robot interactions using potential fields or social forces. Social Force Models (SFM), introduced by Helbing and Molnar [8], simulate human motion as a result of attractive and repulsive forces. Ferrer et al. [9] extended this model to robots, allowing them to navigate crowded environments by anticipating human trajectories and applying proactive strategies to avoid collisions.

The Artificial Potential Field (APF) method is another force-based technique commonly used in robot navigation. APF generates attractive forces toward the goal and repulsive forces away from obstacles. Araujo et al. [10] applied APF in socially aware navigation inside offices.

While these force-based models are computationally efficient and effective in real-time, they often struggle in complex environments where human interactions are more dynamic and less predictable.

2.3.3 Learning-Based Approaches

Learning-based approaches have gained traction due to their ability to capture the nuances of human behavior. These methods often leverage machine learning techniques to generate velocity commands directly from sensor inputs without the need for explicit trajectory generation.

Reinforcement Learning (RL) and Deep Reinforcement Learning (DRL) have been widely used to teach robots socially compliant behaviors. For instance, Chen et al. [11] employed DRL to teach robots to navigate crowds by learning human interaction patterns.

Banisetty and Feil-Seifer have presented an interesting learning-based approach [12] consisting in a Unified Socially Aware Navigation (USAN) model that employs a Gaussian Mixture Model to classify the social scenario and determine cardinal objective for each case. once they are determined, a PaCcET (Pareto Concavity Elimination Transformation) local planner is applied, which allows for multi-objective path optimization.

2.4 Global Planning

Global motion planning involves generating high-level plans that guide the robot to its destination. These plans consider long-term goals and interactions, typically focusing on strategic decisions rather than immediate obstacle avoidance. In the case of socially aware path planning, global planners consider high-level social conventions and interactions.

2.4.1 Planning-Based Approaches

Traditional planning-based approaches, such as A* and Dijkstra's algorithm, are widely used for global path planning. However, these methods were not originally designed to handle social interactions. To bridge this gap, researchers have introduced social cost maps that augment traditional planners. One widely adopted implementation in this category is the ROS social navigation layers, which modify the cost map by applying a Gaussian distribution around people, with the cost heightened in the direction in which they are moving. Eirale et al. [13] focused on learning a cost that accounts for high-level social norms, such as avoiding groups of interacting people, using Convolutional Neural Networks (CNNs). Similarly, this thesis will focus on exploiting the advancements in CNNs to generate a unified social layer that respects such norms.

Sampling-based approaches, such as Rapidly-exploring Random Trees (RRT) and Probabilistic Roadmaps (PRM), are also commonly used in global planning. These methods sample the environment to build a tree or graph of possible paths. Vega et al. [14] extended PRM to handle human environments by incorporating dynamic obstacles and social constraints.

2.4.2 Learning-Based Approaches

Learning-based approaches in global planning typically involve training models to predict long-term interactions and plan accordingly. Inverse Reinforcement Learning (IRL) is a popular method that learns social navigation policies by observing human behavior. Vasquez et al. [15] employed IRL to train robots to navigate crowded environments by inferring the social norms followed by humans.

Deep Learning techniques, such as CNNs, are also used for global planning. Pérez-Higueras et al. [16] trained a CNN-based model to imitate human-designed paths.

Chapter 3

Methodology

3.1 Introduction

Traditional robot navigation methods have been remarkably successful in enabling robots to operate in indoor environments. However, these methods often fall short in social contexts where many human robot interactions occur. To address this limitation, the proposed methodology builds upon conventional grid-based path planners by integrating a learned social cost layer that adapts to human activities. This chapter details the architecture and implementation of the work.

3.2 Background

Grid-based path planners, such as A* and Dijkstra's algorithm, represent the environment as a two-dimensional graph, where each vertex corresponds to a cell in the grid map. The cost of traversing from one cell to another is assigned based on both the distance and the desirability of each cell. These planners optimize for the cheapest feasible path, ensuring obstacle avoidance.

3.3 Socially Aware Navigation

The core of the proposed methodology is the introduction of a social cost layer C_S, learned through a deep neural network. This layer reflects social costs and obstacles that are informed by human activities, such as queuing, or group interactions, which traditional planners do not account for. By augmenting the traditional cost map with this social cost component, the robot can plan paths that are not only efficient and safe but also socially acceptable.

3.4 Network Architecture

The neural network used in this work is a U-Net architecture [Figure 3.1], which is particularly well-suited for image-to-image translation tasks. The U-Net architecture has been widely adopted in various domains, initially in biomedical image segmentation [17] for its ability to work with fewer training images, and now, famously, OpenAI employs it in the DALL-E image generation model for iterative image denoising.

3.4.1 Architectural Overview

The U-Net architecture employed here consists of three main components: convolutional layers for feature extraction, max-pooling layers for down-sampling, and up-sampling layers for reconstructing the output at the original resolution. This network structure is designed to transform the input grid map into a social cost layer, where each pixel represents the social cost associated with that location.

Encoder

The encoder part of the U-Net is responsible for progressively down-sampling the input to capture higher-level features at multiple scales. It consists of a series of convolutional blocks, each followed by a max-pooling operation. Each convolutional block includes two convolutional layers, batch normalization, and ReLU activation. The purpose of these layers is to extract features at different levels of abstraction while maintaining the spatial coherence of the input data.

The use of batch normalization helps stabilize training by normalizing the inputs of each layer, while ReLU activation introduces non-linearity, allowing the network to learn more complex patterns. Max-pooling layers reduce the spatial dimensions, enabling the network to learn more global features as the depth increases.

Decoder

The decoder part of the U-Net is tasked with up-sampling the feature maps back to the original resolution while combining information from higher and lower layers. The decoder consists of up-convolution blocks (also known as transposed convolutions), which increase the spatial dimensions of the feature maps. Each up-convolution block is followed by a concatenation operation with the corresponding feature maps from the encoder, allowing the network to combine coarse, high-level features with fine, low-level features.

This skip connection mechanism is a key innovation of the U-Net architecture, as it enables the network to retain spatial details that might otherwise be lost during down-sampling and improves gradient flow in the backpropagation process.

The concatenated feature maps are then processed by additional convolutional layers, which refine the combined features to produce the final output.

Final Convolution Layer

The final layer of the network is a 1x1 convolution, which reduces the number of channels to one, corresponding to the desired output: the social cost map. This layer essentially collapses the information from all previous layers into a single output map, which is then used for further processing in the navigation pipeline.

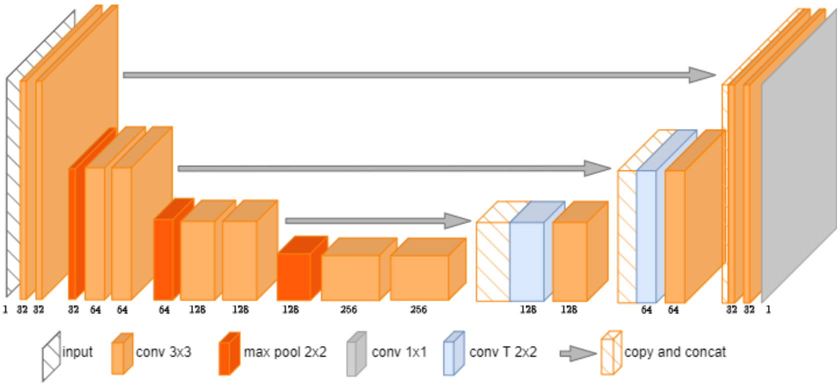


Figure 3.1: Neural network architecture

3.5 Dataset Creation

To train the neural network, the generated dataset consists of input-target 40×40 pixels grayscale image pairs, where the input is a grid map that includes information about people, obstacles, and the robot goal, and the target is the corresponding social cost target.

People's positions are represented by a pixel value of 100, and the orientation is represented by one of the neighboring eight pixels having a value of 25. Obstacles are represented with a pixel value of 255, free space with a pixel value of 0, and the goal is represented with a pixel value of 200. The robot's position is always in the image center and it is not represented in the image.

In the target image, pixel values directly represent cost, so a pixel value of 0 is the lowest possible cost, and a pixel value of 255 is the highest possible cost.

Dataset design is paramount in this supervised scenario, as it directly impacts the robustness of the social cost layer in complex navigation scenarios. The dataset is composed of almost 30 thousand input-target image pairs, 70% of which are utilized for training, while the remaining for validation. In the following subsections, the instances considered are presented alongside images reporting instances of the validation output of the neural network, their respective input, and the relative target [Figures from 3.2 to 3.6].

3.5.1 Corner Cost

For human safety's sake, when the robot approaches a corner, it should maintain at least a one meter distance from the walls, such that if a human unexpectedly appears from behind the corner, a likely collision is spared.

To obtain such a behavior, instances of corners are generated [Figure 3.2]. The input image represents two walls that meet in a corner, while the target represents the corresponding cost, which is 255 at the corner location, and exponentially decays with distance.

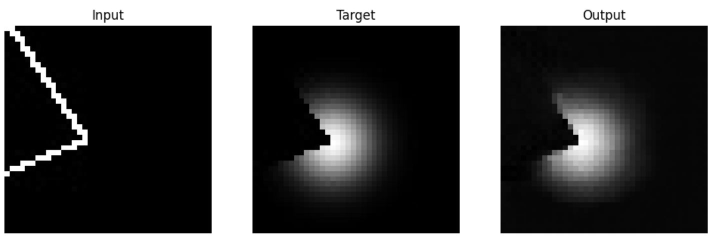


Figure 3.2: Corner scenario

3.5.2 Hallway Cost

In the hallway navigation scenario, the robot is trained to adhere to the social convention of keeping to the right while navigating through hallways.

The training dataset accounts for a generally oriented hallway, with walls of general length, one wall may be shorter than the other [Figure 3.3].

The target is generated keeping into account the robot being positioned in the middle of the image, and such that the keep right social rule is respected when it navigates in either direction. The highest cost is placed close to the left wall, and since it is not safety critical it has a value of 200, and it exponentially decays with distance. if the wall ends within the input image an additional cost is added around the corner, like in the corner cost instance

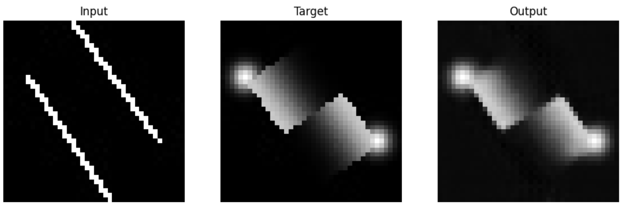


Figure 3.3: Hallway scenario

3.5.3 Single Human

When moving around humans the robot should keep proxemics into account. The input includes the human pose and a randomly generated wall. The corresponding target includes a few asymmetric Gaussian costs around the human position [Figure 3.4], implemented in a very similar fashion as the social navigation layers package of ROS1, in this work the proxemics layer and the passing layer are merged.

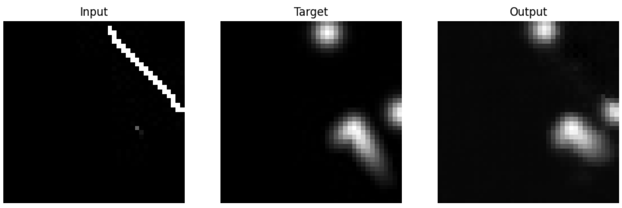


Figure 3.4: Single human scenario

3.5.4 Queuing Scenario

In the queuing scenario, the robot must respect the queue to get to the objective without cutting ahead. In the input image, there is a line of people standing in front of the robot goal. The target consists of a U-shaped cost of 255 [Figure 3.5], which will guide the robot at the end of the queue.

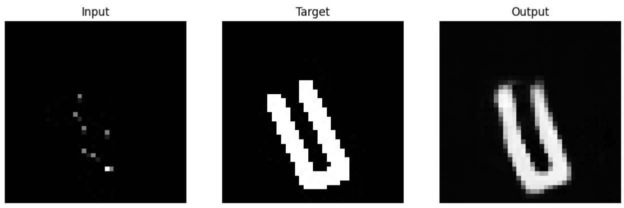


Figure 3.5: Queue scenario

3.5.5 Non-queuing Scenario

In the non-queuing scenario, we have people standing in a queue, but the robot goal is not at the front of the queue. So in this instance, the target image will only include the proxemics cost of humans [Figure 3.6].

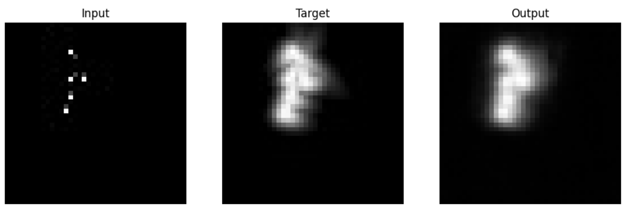


Figure 3.6: Non-queue scenario

3.5.6 Groups of People

When people are staying in groups, the robot is expected to navigate around them, instead of passing through the group. In the input image, people are standing in a group configuration, while in the target image, human proxemics are included, as well as an additional cost of 255 in the group interaction area [Figure 3.7].

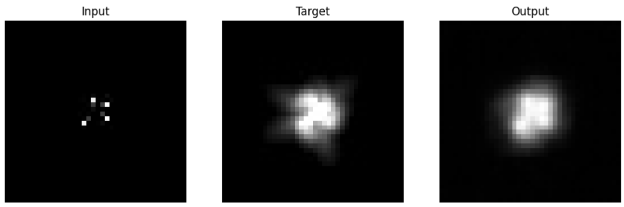


Figure 3.7: Group scenario

3.5.7 Non-groups of People

This instance is generated to facilitate discrimination between groups of people, and several individuals, each standing on their own. In the input image, people are positioned in a classical group configuration, but the orientation is towards the outside [Figure 3.8]. In the target image, only human proxemics are considered.

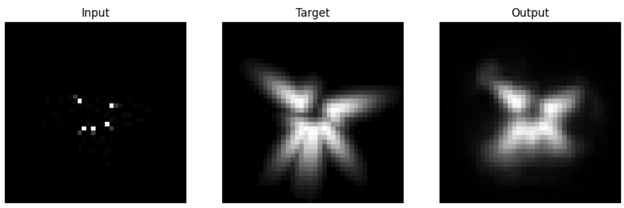


Figure 3.8: Non-group scenario

3.6 ROS2 NAV2 Interface

In this section the social cost layer integration into the ROS2 NAV2 framework, which is the navigation stack used for mobile robotics, is discussed.

3.6.1 Implementation of the Social Cost Layer

The social cost layer is incorporated into the NAV2 costmap through a custom plugin that subscribes to a ROS2 topic, over which is published the social cost layer.

Social Cost Layer Publisher Node

A dedicated ROS2 Python node performs inference to generate the social cost layer and publishes it over a topic to make it available to the custom NAV2 plugin. This node operates as follows:

- *Occupancy Grid Map Processing:* The map is retrieved from the */map* topic, which provides an occupancy grid map message. This message is first converted into a NumPy array for further processing. Each value of the array is an occupancy probability in the range 0 to 100.
- *Map Value Conversion:* In the NumPy array, cells with values above 65 are considered as walls and are converted to 255, while cells with values below 65 are treated as free space and set to 0.
- *Map Rescaling:* The processed map is then rescaled to a resolution of 0.4 meters per cell, which is the resolution used to train the encoder-decoder network. This rescaling ensures consistency between the map data and the network's input requirements.
- *Incorporation of Human Presence:* The positions and orientations of people within the environment are added to the map and updated every time new information is available. Each person is represented as a cell with a value of 100 at their location, and one of the eight neighboring cells is assigned a value of 25 to indicate their orientation. In a real-world scenario, information about people would be obtained from a dedicated perception pipeline, while in a simulated scenario, it can also be directly obtained from the human simulator.
- *Extraction of Local Map Segment:* After incorporating human data, a 16m x 16m square segment of the map, centered around the robot's current position, is extracted from the NumPy array. In a real-world situation, the perception pipeline is also based on lidar; under the assumption of a lidar range of 8m, we have people pose information in a circle of radius 16m, hence the 16m square.

- *Neural Network Inference:* The extracted map segment is then fed into the encoder-decoder neural network. The network processes this input and outputs a corresponding social cost layer, which has the same size as the input segment.
- *Tresholds:* To get rid of background noise, the grid elements with values below 50 are set to zero. While elements above the 195 threshold are set to 255, which is interpreted as a lethal obstacle by NAV2. The two thresholds are obtained through a trial and error process.
- *Social Cost Layer Assembly:* The processed output is then used to assemble a new */OccupancyGrid* message, with a data field that represents the social cost layer, composed of zeros everywhere, except for the 16m square around the robot, where the processed output is placed. This is the social cost layer.
- *Publishing the Social Cost Layer:* Finally, the newly assembled */OccupancyGrid* message, representing the social cost layer, is published on a ROS2 topic. This makes the social cost layer available to the NAV2 plugin, allowing the planner to incorporate social considerations into its path planning process.

Chapter 4

Experimental Setup and Simulation Environments

4.1 Introduction

This chapter presents the experimental setup and simulation environments used to evaluate the capabilities of respecting the social conventions of this unified system. The purpose of these experiments is to test the proposed approach's effectiveness in various scenarios that mimic real-world situations and assess the robustness of the developed neural network. From elementary social scenarios to more complex instances where several conflicting social rules should be considered. The chapter is organized as follows: first, the simulation environment and its components are described in detail; next, the experimental setup, including the hardware and software configurations, is discussed; finally, the specific scenarios used for testing the system are outlined.

4.2 Simulator Selection

For this work, the *Robot Operating System 2 (ROS2)* coupled with the *Gazebo* simulator was selected due to their widespread use in robotics research and their robust support for simulating both the physical dynamics of robots and the complex interactions with dynamic elements like humans. To simulate realistic human-robot interactions, the *HuNavSim* [**PerezRal2023**] plugin was integrated into the simulation environment.

- **ROS2:** ROS2 serves as the middleware, providing the framework for robot control, communication between different modules, and integration with various sensors and actuators.

- **Gazebo:** Gazebo is the simulation engine used to create and simulate the virtual environment. It supports accurate physics-based modeling of robot dynamics and interactions with the environment.

- **HuNavSim:** HuNavSim is a ROS2-based simulator designed for simulating human navigation in environments shared with robots. It allows for the creation of dynamic scenarios where virtual humans move based on predefined or randomized behaviors, providing a realistic context for testing socially aware navigation strategies.

4.3 Human Agents

To simulate realistic human-robot interaction, human agents were modeled within the simulation, thanks to HuNavSim. These agents exhibit behavior patterns that mirror real human movements, including walking, stopping, and avoiding other humans or the robot. The pose of each human agent is directly obtained from HuNavSim, which publishes it over the */people* ROS topic. This allows testing the system without the need for a people detection pipeline.

4.4 Experimental setup

4.4.1 Hardware Configuration

The experiments were conducted on a workstation equipped with the following specifications:

- **Processor:** Intel Core i7-9700K, 8 Cores @ 3.6 GHz

- **Memory:** 32 GB DDR4 RAM

- **Graphics:** NVIDIA GeForce RTX 2080, 8 GB VRAM

- **Storage:** 1 TB NVMe SSD

- **Operating System:** Ubuntu 22.04 LTS

This hardware setup provided enough computational power to run ROS2 and the Gazebo simulations at real-time speeds, allowing for efficient testing and iteration of the path planning algorithms.

4.5 Software Configuration

The software stack used for the experiments included the following components:

- **ROS2 Humble:** The ROS2 Humble distribution was used as the middleware framework.
- **NAV2:** The NAV2 stack was employed for navigation, offering a flexible and powerful framework for robot path planning and control within ROS2.
- **Gazebo 11.10:** The Gazebo 11.10 simulation engine was used to create and run the virtual environments, offering realistic physics simulation.
- **Python 3.10:** Custom scripts were developed in Python 3.10 to implement the data generation process.
- **PyTorch:** PyTorch was used for designing, training, and fine-tuning the U-net responsible for generating the social cost layer.
- **OpenVINO:** The trained neural network model was converted to OpenVINO for optimized deployment, enhancing inference speed and efficiency on the target hardware.

4.6 Simulation Scenarios

A series of simulation scenarios were designed to evaluate the performance of the socially aware path-planning system. These scenarios were selected to represent common real-world challenges that a socially aware robot would encounter.

First, the elementary scenarios are tested, and then more intricate instances, where different social conventions suggest different paths, are analyzed. This way, robot decision-making tradeoffs can be observed.

Since the network learned exceptionally well all the various scenarios, at this step the proposed environments will introduce some intricacies, like walls with different thicknesses, to see how robust the network is to these subtle changes.

4.6.1 Elementary Scenarios

Human Interaction

Objective: To test the robot's ability to navigate around a human.

- *Setup:* The robot is placed in an open area close to a human agent.
- *Challenges:* The robot must account for proxemics and navigate at a safe distance, without making the human uncomfortable.

Empty Hallway Navigation

Objective: To test the robot's ability to navigate an empty hallway while adhering to the social convention of keeping to the right.

- *Setup:* A 3-meter wide and 30-meter long empty hallway is set. One wall is thicker than the other.
- *Challenges:* The robot must demonstrate its adherence to the "keep right" convention, even in the absence of other agents, and it has to show robustness to varying wall thicknesses, ensuring that it follows the social norm consistently.

Corner Navigation with No Humans

Objective: To evaluate the robot's ability to plan around a corner with increased caution, even when no humans are present.

- *Setup:* Three main environments are taken into account: one includes a wall with a 90-degree corner the second one includes a 3 meters wide, L-shaped hallway and the last one is a 3 meters wide T-shaped hallway. No human agents are present in these scenarios. Also in this scenario, the walls present different thicknesses.
- Challenges: The robot must navigate the corner cautiously, taking a wider path around the corner to avoid collision with a potential, unseen human. Additionally, in scenarios with corners in hallways, the robot must robustly place the hallway cost, as previously discussed, regardless of corners or walls of different thicknesses.

Approaching and Avoiding Groups

Objective: To evaluate the robot's capability to recognize and navigate around groups of people standing in an open space, rather than cutting through the group.

- *Setup:* An open area is populated with a group of 6 human agents, engaging in conversation.
- *Challenges:* The robot must identify the group formation and choose paths that go around these groups, avoiding socially inappropriate behaviors such as splitting the group or passing too close to them.

Queueing Behavior

Objective: To assess the robot's ability to recognize and join a queue of people waiting to reach a common goal, such as an entrance or service desk, without attempting to bypass the queue.

- *Setup:* A queue of 4 human agents is formed in front of a simulated service desk. The robot's goal is also the service desk, so it must join the end of the queue and wait for its turn.
- *Challenges:* The robot must accurately detect the queue, find its appropriate place at the end, and avoid any behaviors that might be considered queue-jumping or disruptive.

4.6.2 Intricate Scenarios

Complex Indoor Environments Navigation

Objective: To assess the robot's ability to safely navigate through complex indoor environments that include crowded hallways, busy narrow passages, corners, open spaces, dynamic human agents, and group formations, while adhering to social norms.

- *Setup:* Two main gazebo environments are considered, consisting of a connected series of hallways, corners, open spaces, and narrow passages, simulating a typical indoor setting such as an office or a shopping mall. Some human agents are walking through hallways, passages, and open spaces, while others are standing in groups.
- *Challenges:* The robot must plan through these complex and intricate environments, which are far more complex than situations seen during the training phase. The robot will have to break some of the conflicting social rules to follow others and show robustness towards the combination of different scenarios.

4.7 Evaluation Metrics

4.7.1 Rationale for Human-Centered Evaluation

In the domain of socially aware Navigation, traditional navigation metrics often fall short in capturing the nuanced social expectations and norms that are critical for human-robot interaction. While algorithmic metrics for social navigation do exist, their main advantage is that they serve as a cheap proxy for surveyed metrics in large studies. [18]

Global path planning, unlike local path planning, is more concerned with the overall route a robot takes rather than moment-to-moment interactions. The key consideration is whether the robot's planned path respects social conventions, such as not cutting through groups, and adhering to social norms like "keeping right" in a hallway and not cutting in front of a queue. The robot's adherence to these social expectations is hard to quantify mathematically, and thus, algorithmic quantitative metrics may not fully capture the effectiveness of a socially aware navigation system.

Given these considerations, this thesis adopts a human-centered survey approach to evaluate the robot's planned paths. Human evaluators, representing the end-users of these robotic systems, provide insights into how socially appropriate and acceptable the robot's paths are. This approach ensures that the robot's navigation is aligned with human expectations, which is crucial for its successful integration into shared environments.

4.7.2 Human Evaluation Process

To assess the social appropriateness of the robot's planned paths, a structured human evaluation process is implemented. The following steps outline this process:

Evaluator Selection

A group of 5 evaluators is selected to provide feedback on the robot's paths. This group includes individuals from different demographic backgrounds, varying in age, gender, and familiarity with robotic systems. The inclusion of both individuals familiar with social navigation norms and those with no prior exposure ensures a comprehensive evaluation of the robot's path planning.

Presentation of Planned Paths

The robot's planned paths are presented to the evaluators. Each scenario showcases the robot planning in the aforementioned instances. The presentation illustrates the gazebo environment, the robot's planned trajectory, and the surrounding human agents.

Evaluation Criteria

Evaluators are asked to rate the robot's paths based on the following criteria, using a Likert scale from one to five:

- **Social Compliance:** How well does the robot's path adhere to expected social norms (e.g., maintaining appropriate distances, not cutting through groups)?

- **Naturalness:** Does the robot's path appear natural and comfortable, or does it seem awkward or forced?
- **Safety:** Does the path feel safe, or do you anticipate any potential conflicts with humans?
- **Predictability:** Is the robot's behavior predictable, or did it take any unexpected or confusing path?
- **Comfort:** If you were present in the environment, how comfortable would you feel with the robot following the planned path?

Chapter 5

Results and Analysis

5.1 Introduction

This chapter presents an analysis of the results obtained from the experimental scenarios outlined in Chapter 4. The effectiveness of the proposed socially aware path planning approach is evaluated based on the performance metrics defined earlier.

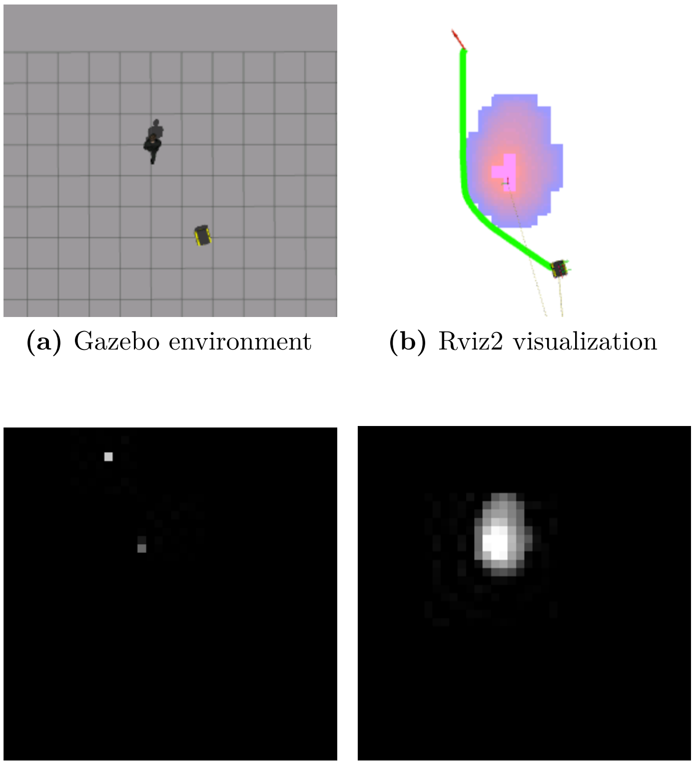
First, the elementary scenarios's results will be briefly analyzed, and then the complex scenarios will be discussed in depth as they are the most significant and better resemble the real world.

5.2 Elementary Scenarios Performance

5.2.1 Scenario 1: Human Interaction

Results: Figure 5.1 reports the results of the single human experiment. The simulation environment can be observed in subfigure (a), where a human is walking in front of the robot. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (a).

Analysis: The robot recognized the human and correctly placed the proxemics cost around the human, which enabled socially compliant planning: the robot avoided the human by passing behind him at a comfortable distance, as the scores show [Figure 5.8 (a)].



(a) Gazebo environment **(b)** Rviz2 visualization

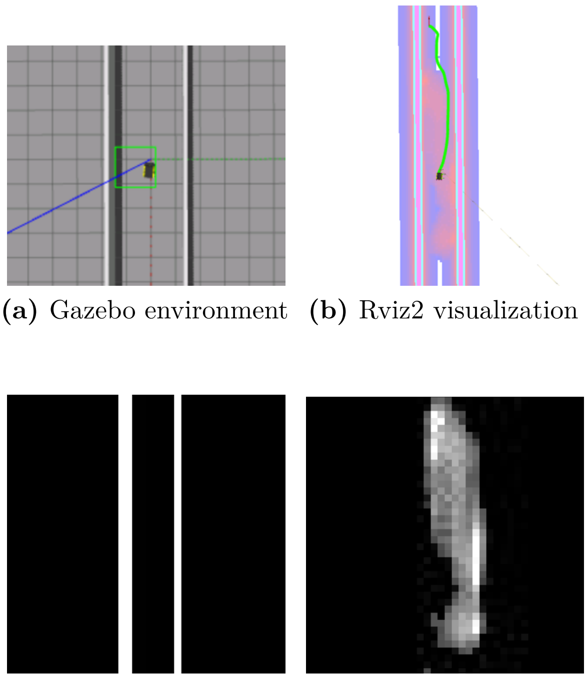
(c) Neural Network input **(d)** Neural Network output

Figure 5.1: Single human instance

5.2.2 Scenario 2: Empty Hallway

Results: Figure 5.2 reports the results of the empty hallway experiment. The simulation environment can be observed in subfigure (a), where the robot is placed in the middle of the hallway. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (b).

Analysis: In this instance the cost map also shows the inflation layer around the walls, it is intended to keep the robot away from the wall. The 16 *times* 16 meters area influenced by the robot can be appreciated in the cost map, in this area, the robot correctly plans to keep right in the hallway. Due to the difference in wall thicknesses, the output of the neural network is not as precise as in the training scenarios, but it still enables the robot to correctly plan its path showing robustness to this kind of unaccounted anomaly. The scores confirm a high degree of social compliance [Figure 5.8 (b)].



(a) Gazebo environment **(b)** Rviz2 visualization

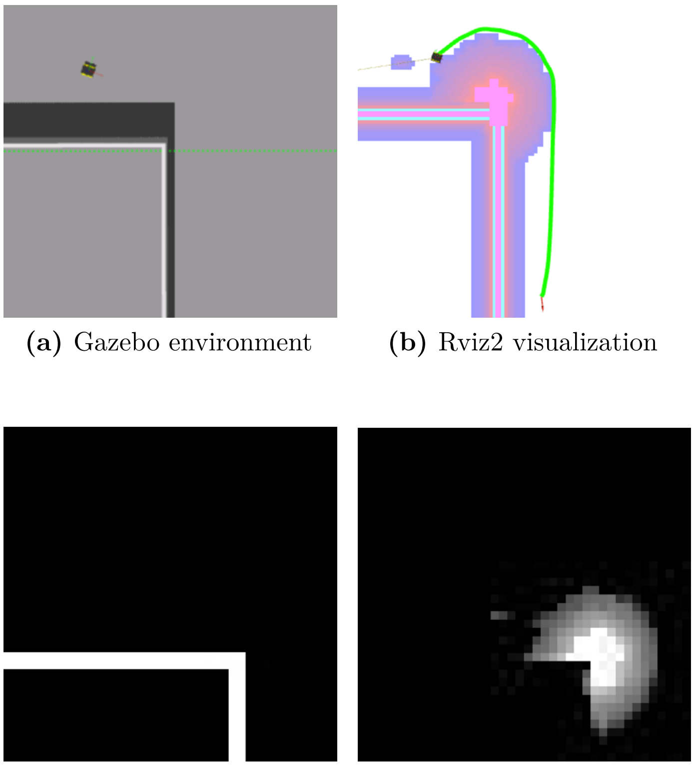
(c) Neural Network input **(d)** Neural Network output

Figure 5.2: Hallway instance

5.2.3 Scenario 3: Single Corner

Results: Figure 5.3 reports the results of the single corner experiment. The simulation environment can be observed in subfigure (a), where the robot is in proximity to the corner. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (c).

Analysis: Also in this case the cost map shows the inflation layer, yet it is evident the presence of the cost due to the social cost layer, which makes the robot correctly plan a wide trajectory around the corner, enhancing human safety in the shared human-robot environment. In this case, the metrics highlight a lower naturalness compared to the other scenarios, this is most likely due to the unnaturalness associated with taking a wide corner. Since the decrease is to allow for enhanced safety, it is acceptable.



(a) Gazebo environment **(b)** Rviz2 visualization

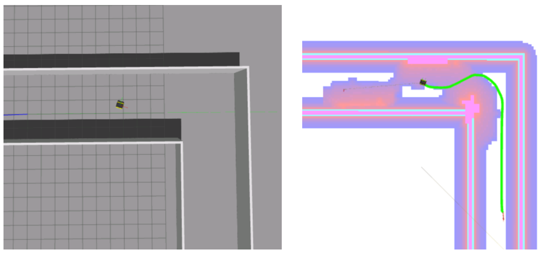
(c) Neural Network input **(d)** Neural Network output

Figure 5.3: Single human instance

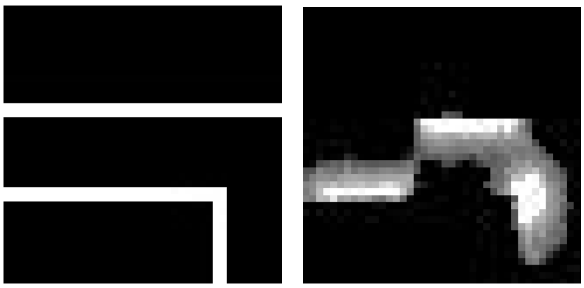
5.2.4 Scenario 4: L-Shaped Hallway

Results: Figure 5.4 reports the results of the L-shaped experiment. The simulation environment can be observed in subfigure (a), where the robot is in proximity to the hallway corner. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (d).

Analysis: Also in this instance, the cost map shows the inflation layer, yet it is evident the presence of the additional cost due to the social cost layer around the corner and along the hallway, which makes the robot correctly plan a wide trajectory around the corner. Due to the combination of the hallway cost and corner cost, the plan is to first co to the right, in the direction of the corner, to respect the hallway cost. Then, to guarantee safety the path steers the robot to its left, this behavior implies naturalness and predictability scores, slightly lower than the other instances.



(a) Gazebo environment **(b)** Rviz2 visualization



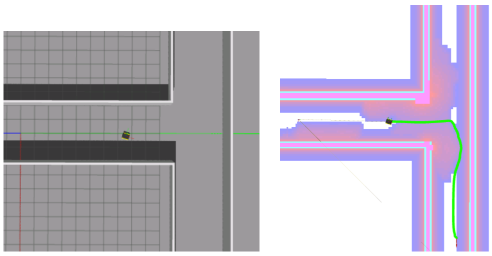
(c) Neural Network input **(d)** Neural Network output

Figure 5.4: Single human instance

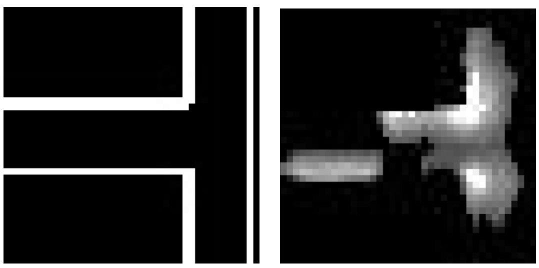
5.2.5 Scenario 5: T-Shaped Hallway

Results: Figure 5.5 reports the results of the T-shaped experiment. The simulation environment can be observed in subfigure (a), where the robot is in proximity to the hallway corner. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (e).

Analysis: Like in the previous scenarios, the cost map shows the inflation layer, yet it is evident the presence of the additional cost due to the social cost layer around the corner and along the hallway, which makes the robot correctly plan a wide trajectory around the corner. In this instance, due to the corner cost being on both sides of the hallway, the planned path is straighter, and it almost reaches the wall. As the robot takes the turn to its right, the wall in question becomes the left side of the hallway, this means that the robot brakes the social rule of keeping right. This is reflected by a social compliance score lower than average, like naturalness and predictability. But, since this is accompanied by high safety, this behavior is expected.



(a) Gazebo environment **(b)** Rviz2 visualization



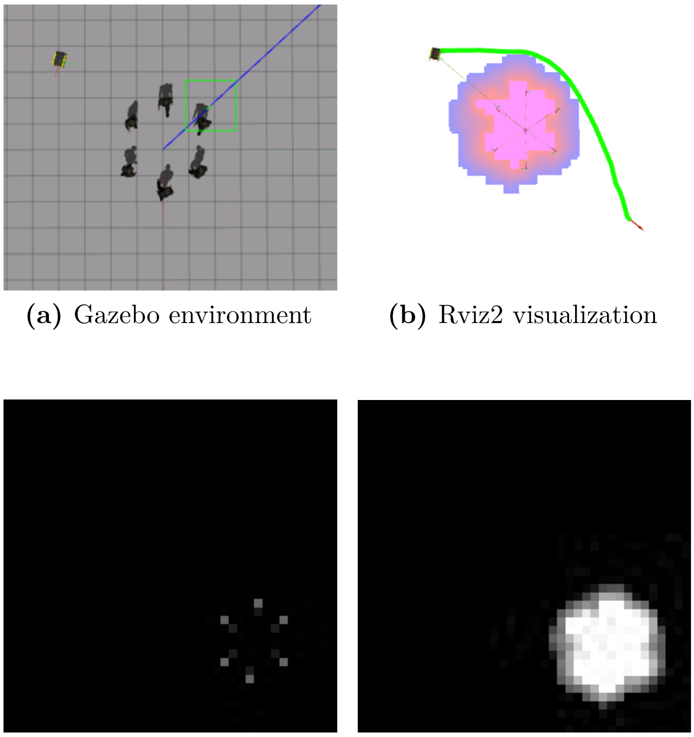
(c) Neural Network input **(d)** Neural Network output

Figure 5.5: Single human instance

5.2.6 Scenario 6: Approaching and Avoiding Groups

Results: Figure 5.6 reports the results of the group instance. The simulation environment can be observed in subfigure (a), where the robot approaches a 6-person group. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (f).

Analysis: The robot successfully assigns both the proxemics and group costs, allowing for comfortable distances and nondisruptive planning around the group. The human evaluation metrics showcase a very high social compliance.



(a) Gazebo environment **(b)** Rviz2 visualization

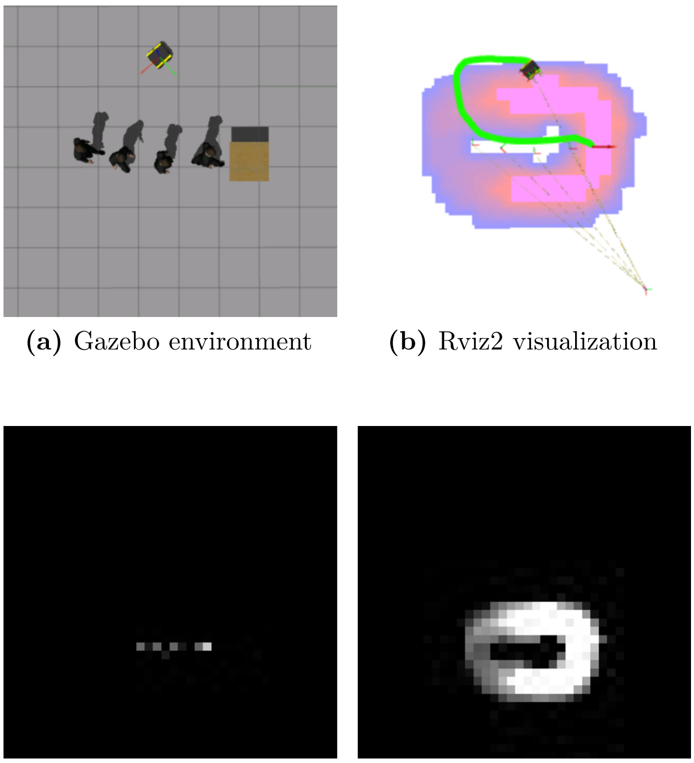
(c) Neural Network input **(d)** Neural Network output

Figure 5.6: Group instance

5.2.7 Scenario 6: Queueing

Results: Figure 5.7 reports the results of the queue instance. The simulated environment can be observed in subfigure (a), where the robot approaches a 4-person queue. In subfigure (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d). Human evaluation metrics are reported in Figure 5.8 (g).

Analysis: Despite the slight cost imperfection at the end of the queue, the robot still plans successfully towards the end of the queue. This is reflected by the good scores assigned by the human evaluators.



(a) Gazebo environment **(b)** Rviz2 visualization

(c) Neural Network input **(d)** Neural Network output

Figure 5.7: Queue instance

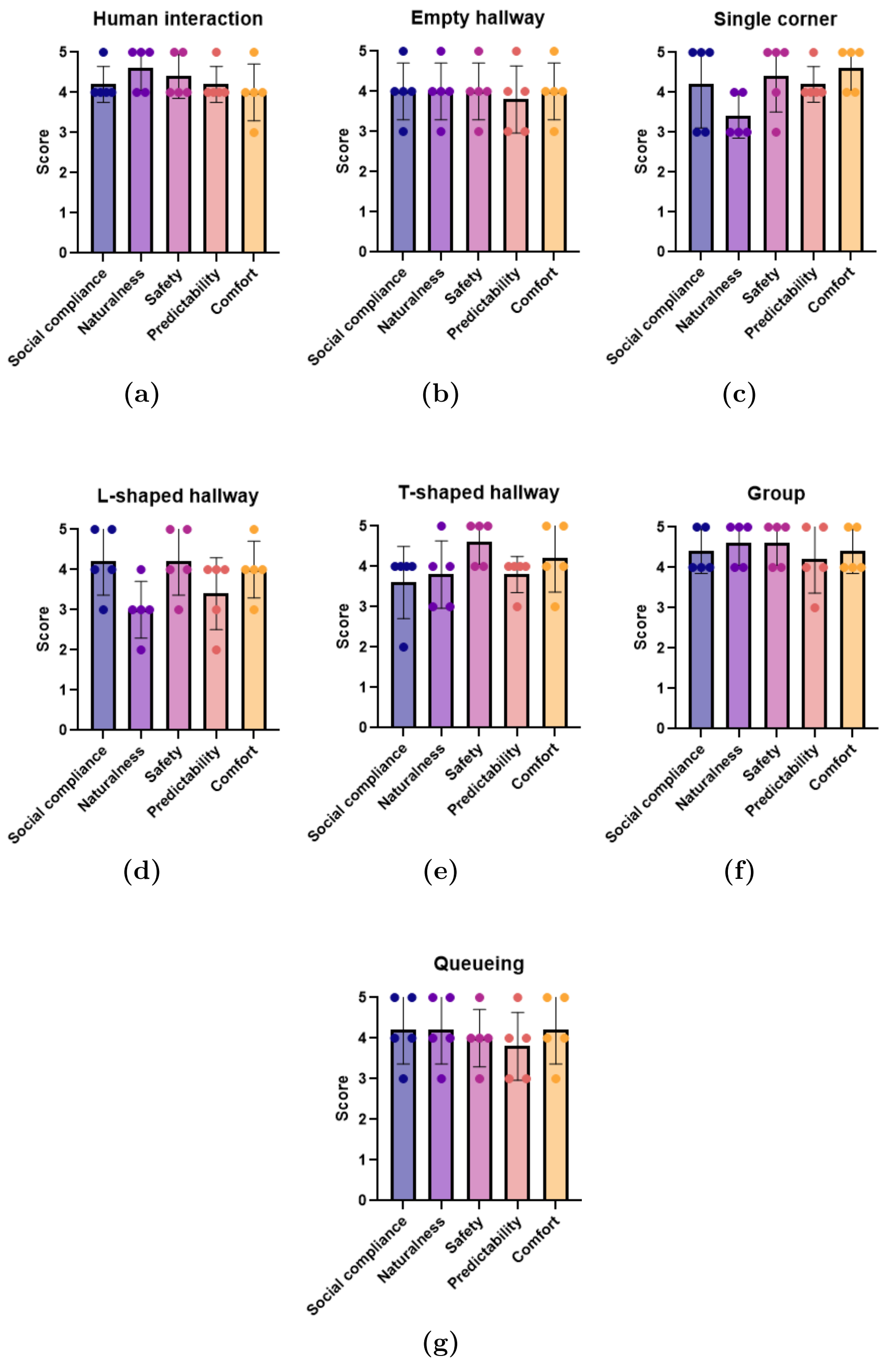


Figure 5.8: Human metrics for the simple scenarios

5.3 Complex Scenarios Performance

In the following complex scenarios, tradeoffs between different social rules can be appreciated. The robustness of the neural network is challenged, both with intricate maps and with highly dynamic human environments.

5.3.1 Scenario 1

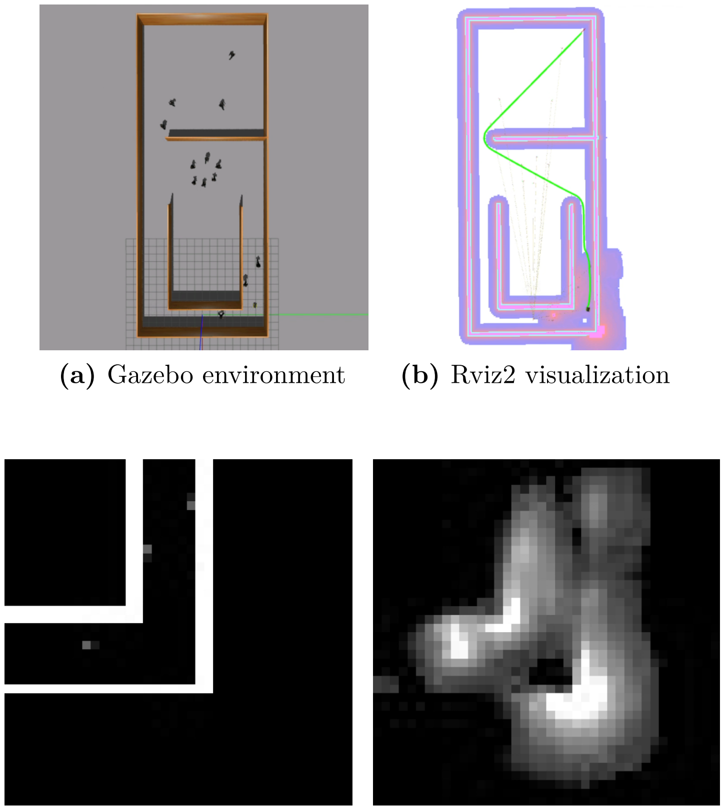
Results: In this scenario the robot navigates first through a populated hallway, which conducts it to a wide open space, where a group of people is interacting. Then it goes through a passage, where a human moves back and forth, reaching a wide room diagonally traversed to reach the goal. Four different relevant time instants are considered, and navigation instances both with and without the social cost layer are reported to highlight the difference between the approaches [Figures from 5.9 to 5.16]. The simulation environment can be observed in subfigures (a). In all the subfigures (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d).

Analysis: There is a significative difference between the navigation with and without the social cost layer, it is apparent since the first time instant [Figures 5.9 and 5.10], where the social cost layer assigns a high cost to the left side of the hallway, and the humans populating it. The planned path respects the right-hand rule for the first few meters of the hallway. But in the area of the human to the right of the hallway, the right-hand rule is not respected, while the proxemics cost is still present [Figure 5.9 (d)]. This is good behavior since an overtake has to be performed, it is better to invade the empty left side of the hallway rather than getting excessively close to the human.

At the second time instant, the group is sighted by the robot and the social cost layer correctly applies the cost [Figure 5.11 (d)], correctly planning around the group [Figure 5.11 (b)].

In the last two time instants, an example of replanning due to the dynamics of the environments is observed in the navigation with the social cost layer. In the third time instant the plan is to pass in front of the moving people [Figure 5.13 (b)], but, as they move on, the plan changes and the robot navigates behind the people respecting the comfort distance.

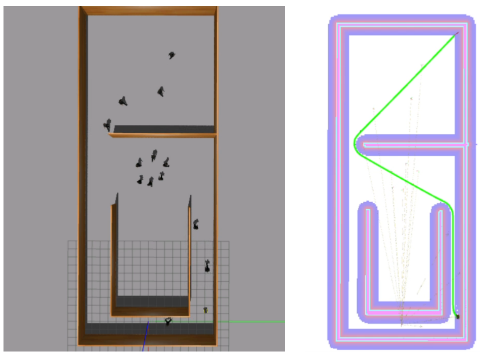
The navigation instance without the social cost layer, does not take social considerations into account, and scores quite low on all the metrics. Especially on safety and social compliance, as the robot navigates very close to people and through the group.



(a) Gazebo environment **(b)** Rviz2 visualization

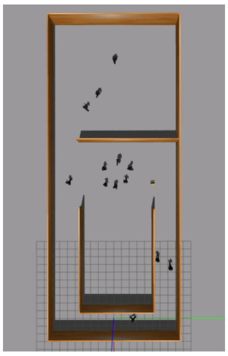
(c) Neural Network input **(d)** Neural Network output

Figure 5.9: Complex Scenario 1, first time instant with social cost layer

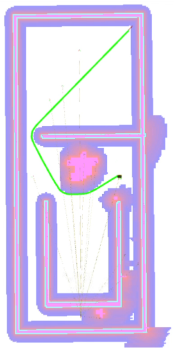


(a) Gazebo environment **(b)** Rviz2 visualization

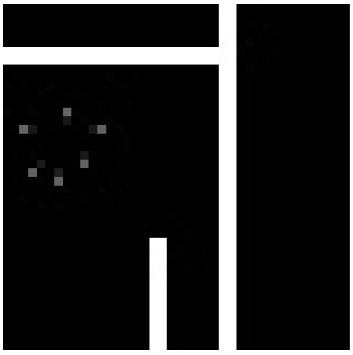
Figure 5.10: Complex Scenario 1, first time instant without social cost layer



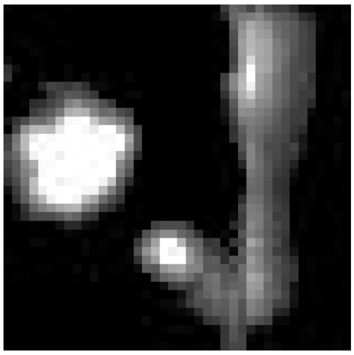
(a) Gazebo environment



(b) Rviz2 visualization

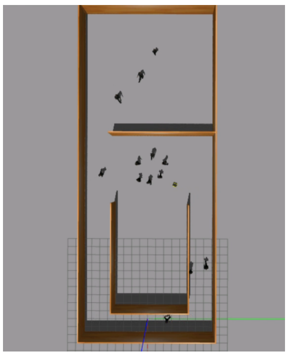


(c) Neural Network input

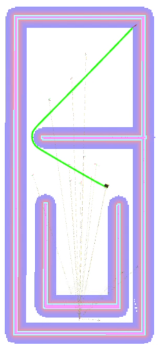


(d) Neural Network output

Figure 5.11: Complex Scenario 1, second time instant with social cost layer

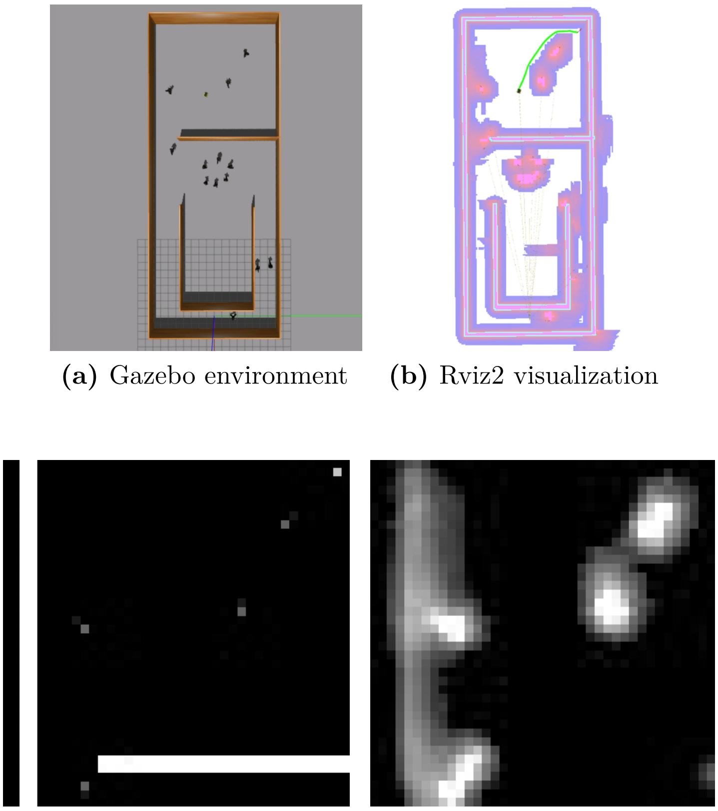


(a) Gazebo environment



(b) Rviz2 visualization

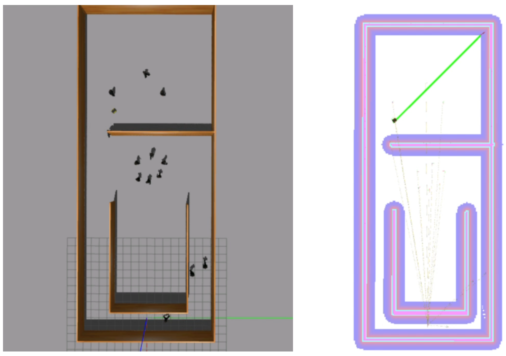
Figure 5.12: Complex Scenario 1, second time instant without social cost layer



(a) Gazebo environment (b) Rviz2 visualization

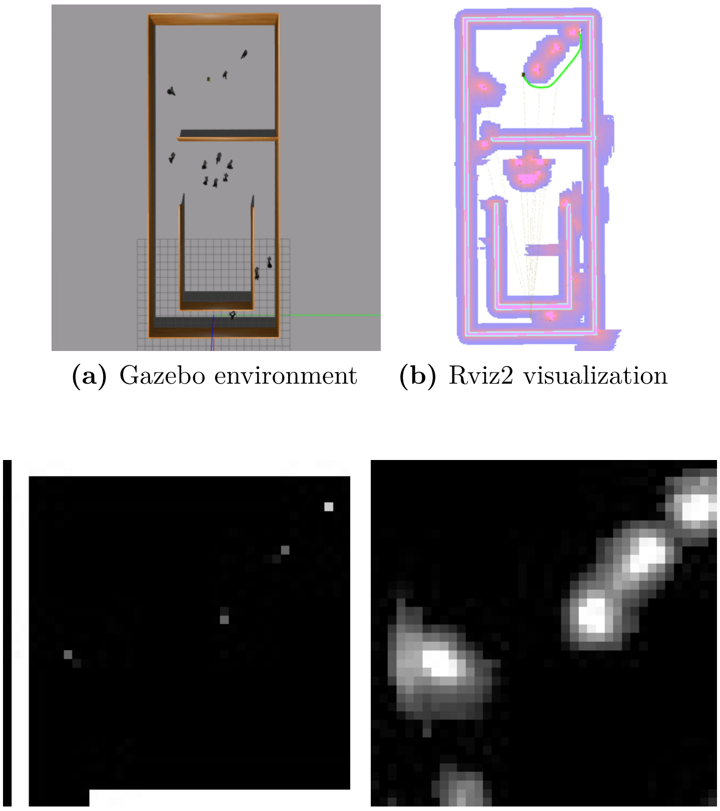
(c) Neural Network input (d) Neural Network output

Figure 5.13: Complex Scenario 1, third time instant with social cost layer



(a) Gazebo environment (b) Rviz2 visualization

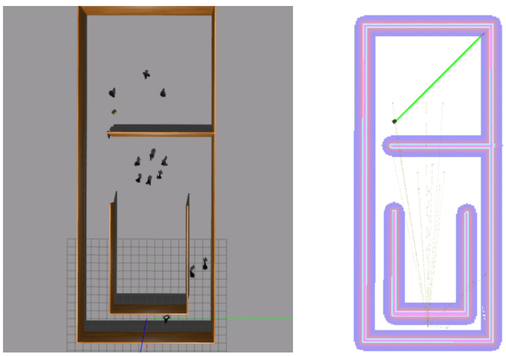
Figure 5.14: Complex Scenario 1, second time instant without social cost layer



(a) Gazebo environment (b) Rviz2 visualization

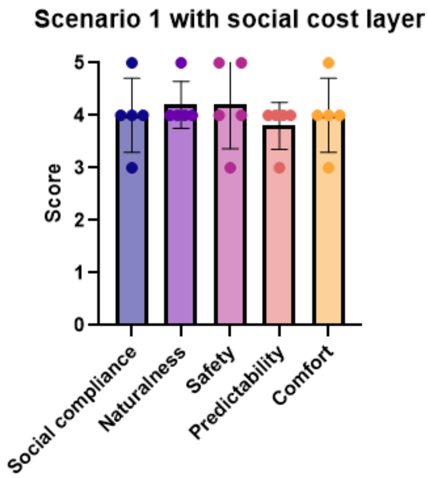
(c) Neural Network input (d) Neural Network output

Figure 5.15: Complex Scenario 1, third time instant with social cost layer

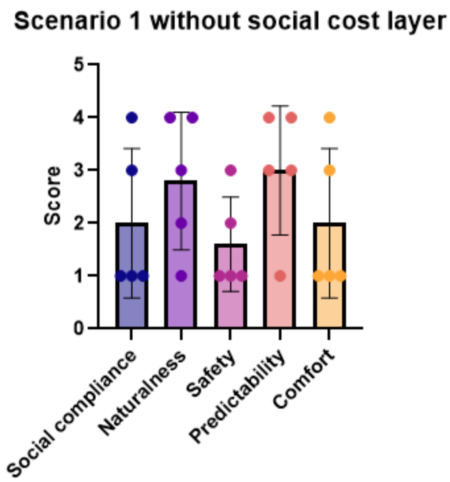


(a) Gazebo environment (b) Rviz2 visualization

Figure 5.16: Complex Scenario 1, second time instant without social cost layer



(a)



(b)

Figure 5.17: Scenario 1 human evaluation metrics

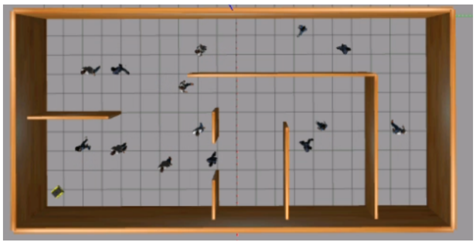
5.3.2 Scenario 2

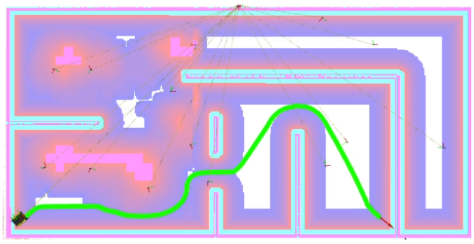
Results: In this scenario the robot navigates through an open space to reach an enclosed space, that is accessible from two different entrances. Two different significant time instants are considered, and navigation instances both with and without the social cost layer are reported to highlight the difference between the approaches, figures from 5.18 to 5.21. The simulation environment can be observed in subfigures (a). In all the subfigures (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d).

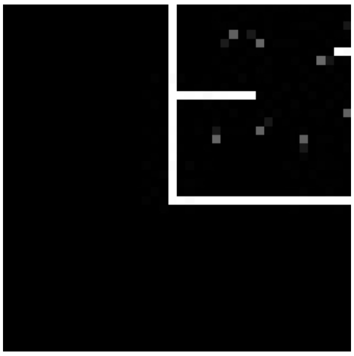
Analysis: In this experiment the social cost layer allowed the robot to select an exceptionally socially compliant behaviour. In the first time instant, the robot plans a path that respects proxemics [Figure 5.18 (b)], the planned path passes through the bottom entrance, which, at this time instant, is more than 8m away, hence it is not seen by the input to the neural network [Figure 5.18 (c)].

At the next time instant, both entrances to the enclosed space are represented in the input image to the neural network [Figure 5.20 (c)]. The bottom entrance is crowded and the output of the neural network highlights it with a high cost in that area. So the robot plans to through the top entrance [Figure 5.20 (b)], which is free of humans. This implies a slightly longer path, but a socially compliant one, this is the right choice expected from a socially compliant planner.

The effectiveness of the social planner is highlighted in the human evaluation metrics, where the planner that does not consider the social cost layer, planning through the bottom entrance, scores significantly lower in all the metrics.


(a) Gazebo environment


(b) Rviz2 visualization


(c) Neural Network input

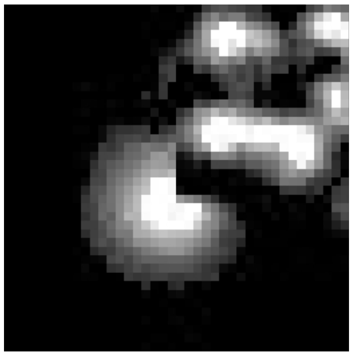
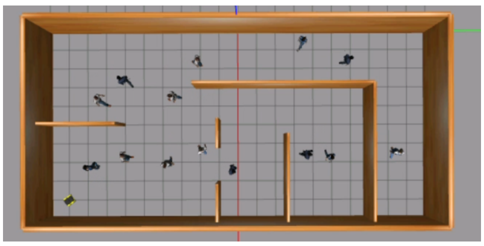

(d) Neural Network output

Figure 5.18: Complex Scenario 2, first time instant with social cost layer


(a) Gazebo environment

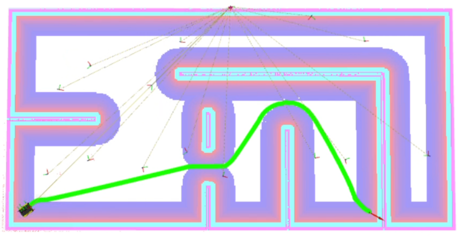
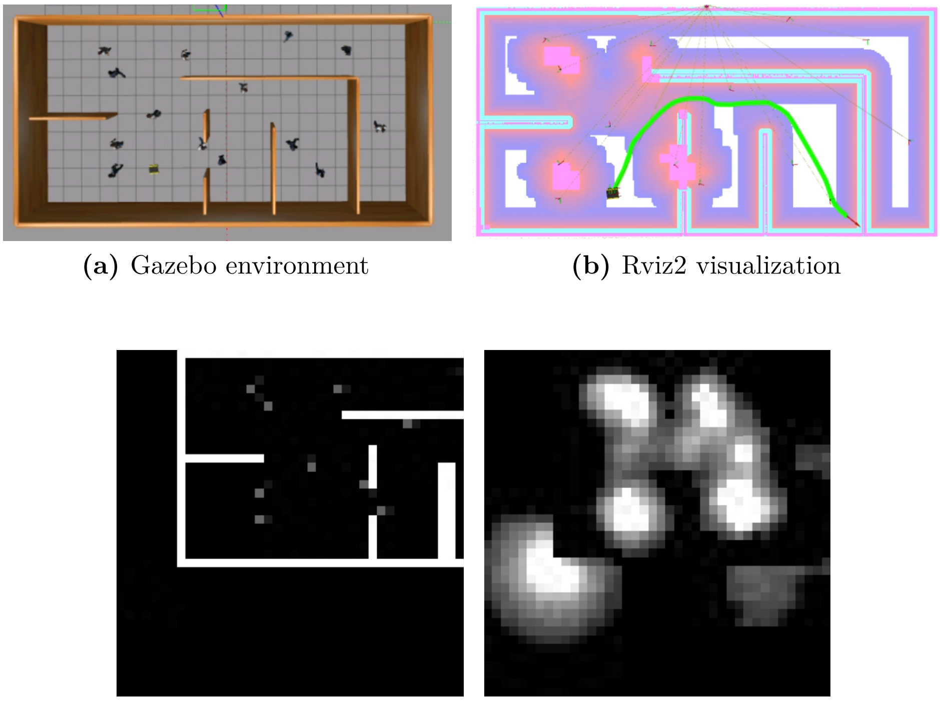

(b) Rviz2 visualization

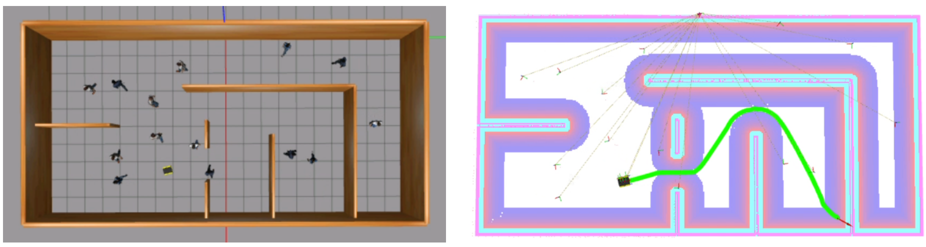
Figure 5.19: Complex Scenario 2, first time instant without social cost layer



(a) Gazebo environment (b) Rviz2 visualization

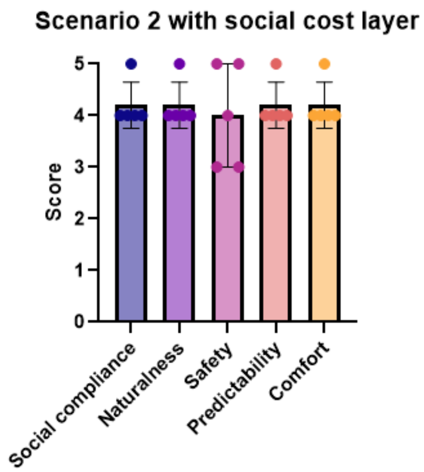
(c) Neural Network input (d) Neural Network output

Figure 5.20: Complex Scenario 2, second time instant with social cost layer

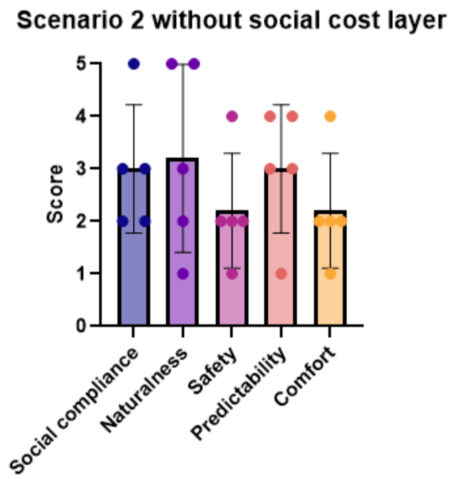


(a) Gazebo environment (b) Rviz2 visualization

Figure 5.21: Complex Scenario 2, second time instant without social cost layer



(a)



(b)

Figure 5.22: Scenario 2 human evaluation metrics

5.3.3 Scenario 3

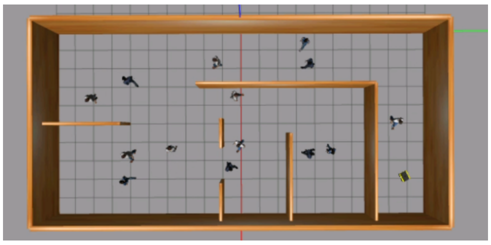
Results: In this scenario the robot navigates in the same environment as in Scenario 2, but the starting position is at the bottom right of the map. Here the robot first navigates through a populated L-shaped hallway, which conducts it to a crowded open space. The goal is set at the bottom left corner of the open space. Two different significant time instants are considered, and navigation instances both with and without the social cost layer are reported to highlight the difference between the approaches, figures from 5.18 to 5.21. The simulation environment can be observed in subfigures (a). In all the subfigures (b) the cost map is visualized with colors ranging from blue, representing low cost, to pink, representing high cost, while the planned path is shown in green. The neural network input and output can also be observed, respectively in subfigures (c) and (d).

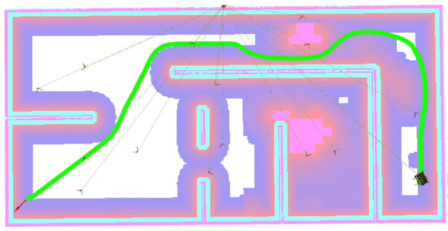
Analysis: In this scenario the planner's ability to dynamically plan around people is showcased. Even though the task of avoiding people in close proximity of the robot is a task concerning social local planners, it is still useful to dynamically avoid humans and respect their personal space with the global planners.

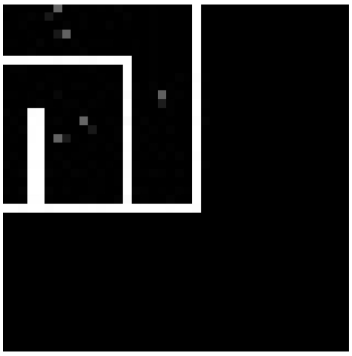
In the first time instant, the social cost layer correctly prioritizes proxemics costs over hallway costs, it is particularly highlighted for the people standing in the horizontal section of the hallway, in the top of the map. The proxemics cost makes the robot plan to the left side of the hallway, which is a socially correct decision to make.

At the second instant, the robot reached the crowded open space. Now, the planned path correctly goes around people in a non-disruptive way, ensuring people's comfort.

The effectiveness of the social cost layer is, once again, confirmed by the evaluation metrics, which present a significant difference in the two navigation scenarios.


(a) Gazebo environment


(b) Rviz2 visualization


(c) Neural Network input

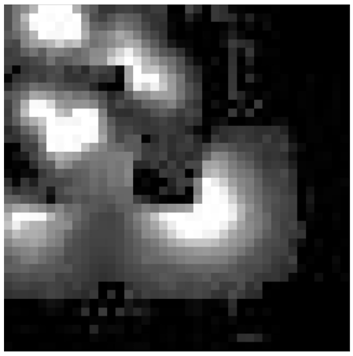
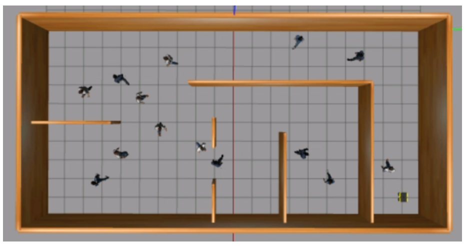

(d) Neural Network output

Figure 5.23: Complex Scenario 3, first time instant with social cost layer


(a) Gazebo environment

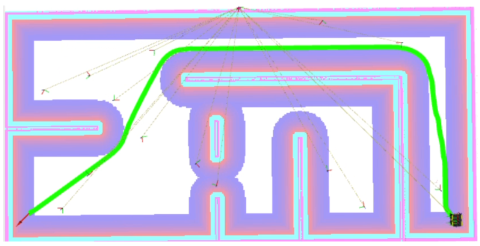
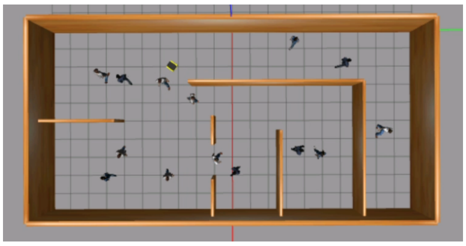
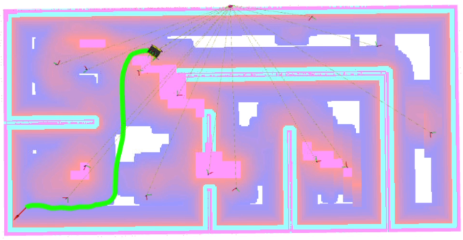

(b) Rviz2 visualization

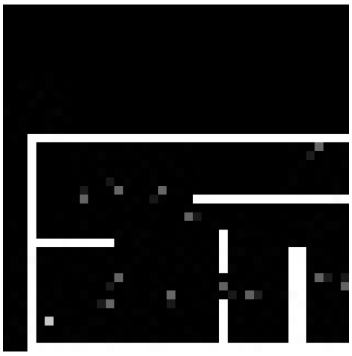
Figure 5.24: Complex Scenario 3, first time instant without social cost layer



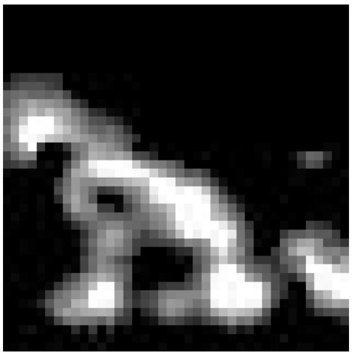
(a) Gazebo environment



(b) Rviz2 visualization

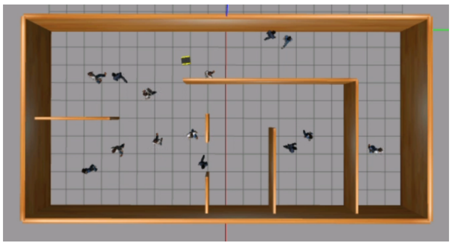


(c) Neural Network input

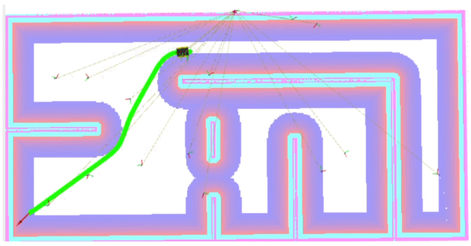


(d) Neural Network output

Figure 5.25: Complex Scenario 3, second time instant with social cost layer

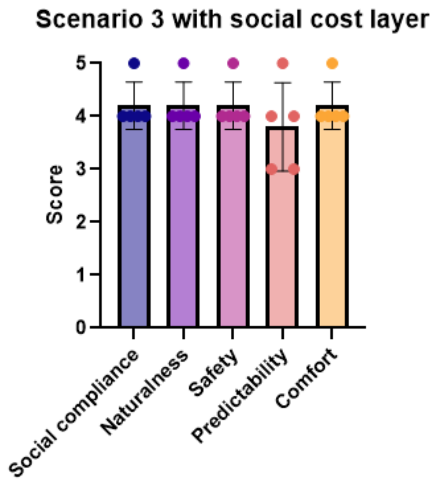


(a) Gazebo environment

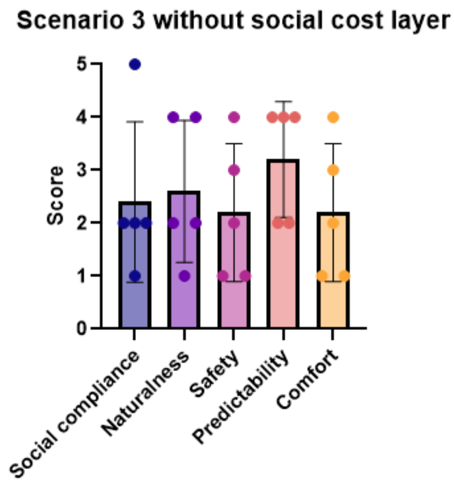


(b) Rviz2 visualization

Figure 5.26: Complex Scenario 3, second time instant without social cost layer



(a)



(b)

Figure 5.27: Scenario 3 human evaluation metrics

5.4 Limitations and Discussion

The in-depth analysis of the results demonstrates that the proposed socially aware path planning system is effective in navigating complex environments while adhering to social norms. The system successfully balances the trade-offs between conflicting social norms, ensuring safe and comfortable human-robot interactions. However, some limitations highlight the need for further research:

- **Scalability:** The current system was tested in small to medium-sized environments. Scaling to larger environments with more complex maps and human interactions may require further optimization of the path planning algorithm to maintain real-time performance.
- **Real-World Implementation:** While the simulation results are promising, real-world implementation may reveal additional challenges, such as sensor noise, slow detection pipeline due to hardware limitations, and the unpredictability of human behavior in uncontrolled environments.
- **Generalization to Diverse Social Norms:** The system was trained and tested in scenarios based on specific social norms (e.g., keeping right in hallways). In environments with different cultural norms, the system might require retraining or adaptation to ensure continued social compliance.

Chapter 6

Conclusion and Future Work

6.1 Introduction

This chapter concludes the thesis by summarizing the key findings and contributions of the research, reflecting on the broader implications of the work, and suggesting potential directions for future research in the field of socially aware navigation for mobile robotics. The goal is to provide a clear and concise overview of the accomplishments of this thesis and to identify areas where further exploration could lead to significant advancements.

6.2 Summary of Findings

The primary objective of this thesis was to develop a socially aware path planning algorithm that allows mobile robots to navigate complex environments while adhering to social norms. The key findings from this research can be summarized as follows:

- **Social Cost Layer Development:** A novel heuristic was introduced for classical path planning algorithms, incorporating a social cost layer that accounts for social conventions such as respecting personal space, keeping to the right in hallways, avoiding groups, and queueing. This layer successfully guided the robot in maintaining socially compliant behaviors in various scenarios.
- **Effective Simulation and Testing:** The developed algorithm was tested in a series of simulation environments designed to mimic real-world social settings. The robot consistently achieved high social compliance scores and maintained human comfort, validating the effectiveness of the approach.
- **Adaptability to Dynamic Environments:** The system demonstrated a strong ability to adapt to dynamic human behaviors, although slight delays

were observed in highly unpredictable scenarios. This adaptability is crucial for real-world applications where human behaviors can change rapidly and unpredictably.

6.3 Contributions

This thesis, with its unified socially aware path planner, brings a significative contribution to the field of socially aware navigation for mobile robotics. The introduction of a social cost layer, generated using a single neural network, allows the robot to evaluate trade-offs between conflicting social norms during the path planning process, resulting in more human-friendly navigation strategies.

6.4 Future Work

While this thesis has made substantial progress in developing and evaluating a socially aware path planning system, several avenues for future research remain open:

6.4.1 Scalability to Larger and More Complex Environments

The current system has been tested in relatively controlled environments. Future work should explore scaling the approach to larger and more complex environments, such as multi-floor buildings, where the robot must navigate more diverse social settings.

6.4.2 Real-World Implementation

Transitioning from simulation to real-world deployment is a critical step. Future research should focus on implementing the proposed system on physical robots and testing it in real-world scenarios. This would involve addressing challenges such as hardware limitations that imply a slow detection pipeline, and the unpredictability of human behavior in uncontrolled environments.

6.4.3 Cultural Adaptability

Social norms vary across different cultures and contexts. Future research could explore how the system can be adapted to respect varying social norms, potentially through training on culturally diverse datasets or incorporating adaptive learning techniques that allow the robot to adjust its behavior based on the social norms of the environment it is operating in.

6.4.4 Integration with Multi-Robot Systems

In environments where multiple robots operate alongside humans, coordination between robots becomes crucial. Future work could investigate how the proposed socially aware navigation system can be extended to multi-robot systems, ensuring that all robots adhere to social norms while avoiding conflicts with one another.

6.5 Broader Implications

The research presented in this thesis has broader implications for the development of socially aware robots capable of interacting safely and comfortably with humans in shared environments. As robots become increasingly prevalent in everyday life, ensuring that they can navigate in a manner that respects human social norms is essential for their acceptance and integration into society.

The proposed approach contributes to this goal by providing a framework for developing robots that are not only effective in their tasks but also considerate of the social dynamics of the spaces they inhabit. This has potential applications in a wide range of settings, from healthcare and service industries to public transportation and personal robotics.

6.6 Conclusion

In conclusion, this thesis has successfully developed and validated a socially aware path planning system for mobile robots, demonstrating its effectiveness in adhering to social norms while navigating complex environments. The contributions of this work provide a solid foundation for future research and development in the field, with the potential to significantly advance the state of the art in socially aware robotics.

As robots continue to be integrated into our daily lives, the ability to navigate in a socially compliant manner will become increasingly important. The findings and contributions of this thesis represent a step forward in this direction, offering new tools and insights that can help shape the future of human-robot interaction.

Bibliography

[1] Wolfram Burgard Sebastian Thrun. «The Interactive Museum Tour-Guide Robot». In: (1997) (cit. on p. 3).

[2] Wolfram Burgard Sebastian Thrun. «MINERVA: A Second-Generation Museum Tour-Guide Robot». In: (1999). URL: https://www.researchgate.net/publication/221606068_The_Interactive_Museum_Tour-Guide_Robot (cit. on p. 3).

[3] E.A. Sisbot, L.F. Marin-Urias, R. Alami, and T. Simeon. «A Human Aware Mobile Robot Motion Planner». In: *IEEE Transactions on Robotics* 23.5 (Oct. 2007), pp. 874–883. ISSN: 1552-3098. DOI: 10.1109/TRO.2007.904911. URL: http://ieeexplore.ieee.org/document/4339546/ (visited on 10/15/2024) (cit. on p. 3).

[4] Viet-Anh Le, Behdad Chalaki, Vaishnav Tadiparthi, Hossein Nourkhiz Mahjoub, Jovin D'sa, and Ehsan Moradi-Pari. *Social Navigation in Crowded Environments with Model Predictive Control and Deep Learning-Based Human Trajectory Prediction*. arXiv:2309.16838 [cs]. Sept. 2023. URL: http://arxiv.org/abs/2309.16838 (visited on 10/15/2024) (cit. on p. 4).

[5] Ha Quang Thinh Ngo, Van Nghia Le, Vu Dao Nguyen Thien, Thanh Phuong Nguyen, and Hung Nguyen. «Develop the socially human-aware navigation system using dynamic window approach and optimize cost function for autonomous medical robot». en. In: *Advances in Mechanical Engineering* 12.12 (Dec. 2020), p. 1687814020979430. ISSN: 1687-8132, 1687-8140. DOI: 10.1177/1687814020979430. URL: https://journals.sagepub.com/doi/10.1177/1687814020979430 (visited on 10/15/2024) (cit. on p. 4).

[6] Maria Kabtoul, Anne Spalanzani, and Philippe Martinet. «Proactive And Smooth Maneuvering For Navigation Around Pedestrians». In: *2022 International Conference on Robotics and Automation (ICRA)*. Philadelphia, PA, USA: IEEE, May 2022, pp. 4723–4729. ISBN: 978-1-72819-681-7. DOI: 10.1109/ICRA46639.2022.9812255. URL: https://ieeexplore.ieee.org/document/9812255/ (visited on 10/15/2024) (cit. on p. 4).

[7] Roesmann Christoph Torsten Bertram. «Trajectory modification considering dynamic constraints of autonomous robots». eng. In: (2012) (cit. on p. 4).

[8] Dirk Helbing and Péter Molnár. «Social force model for pedestrian dynamics». en. In: *Physical Review E* 51.5 (May 1995), pp. 4282–4286. ISSN: 1063-651X, 1095-3787. DOI: 10.1103/PhysRevE.51.4282. URL: https://link.aps.org/doi/10.1103/PhysRevE.51.4282 (visited on 10/15/2024) (cit. on p. 4).

[9] Gonzalo Ferrer, Anais Garrell, and Alberto Sanfeliu. «Robot companion: A social-force based approach with human awareness-navigation in crowded environments». In: *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Tokyo: IEEE, Nov. 2013, pp. 1688–1694. ISBN: 978-1-4673-6358-7 978-1-4673-6357-0. DOI: 10.1109/IROS.2013.6696576. URL: http://ieeexplore.ieee.org/document/6696576/ (visited on 10/15/2024) (cit. on p. 4).

[10] Arthur R. Araujo, Daniel D. Caminhas, and Guilherme A.S. Pereira. «An Architecture for Navigation of Service Robots in Human-Populated Office-like Environments». en. In: *IFAC-PapersOnLine* 48.19 (2015), pp. 189–194. ISSN: 24058963. DOI: 10.1016/j.ifacol.2015.12.032. URL: https://linkinghub.elsevier.com/retrieve/pii/S2405896315026567 (visited on 10/15/2024) (cit. on p. 5).

[11] Yu Fan Chen, Michael Everett, Miao Liu, and Jonathan P. How. *Socially Aware Motion Planning with Deep Reinforcement Learning*. arXiv:1703.08862 [cs]. May 2018. URL: http://arxiv.org/abs/1703.08862 (visited on 10/15/2024) (cit. on p. 5).

[12] Santosh Balajee Banisetty and David Feil-Seifer. *Towards a Unified Planner For Socially-Aware Navigation*. Version Number: 2. 2018. DOI: 10.48550/ARXIV.1810.00966. URL: https://arxiv.org/abs/1810.00966 (visited on 10/15/2024) (cit. on p. 5).

[13] Andrea Eirale, Matteo Leonetti, and Marcello Chiaberge. *Learning Social Cost Functions for Human-Aware Path Planning*. arXiv:2407.10547 [cs]. July 2024. URL: http://arxiv.org/abs/2407.10547 (visited on 10/15/2024) (cit. on p. 5).

[14] Araceli Vega-Magro, Luis Manso, Pablo Bustos, Pedro Nunez, and Douglas G. Macharet. «Socially acceptable robot navigation over groups of people». In: *2017 26th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN)*. Lisbon: IEEE, Aug. 2017, pp. 1182–1187. ISBN: 978-1-5386-3518-6. DOI: 10.1109/ROMAN.2017.8172454. URL: http://ieeexplore.ieee.org/document/8172454/ (visited on 10/15/2024) (cit. on p. 6).

[15] Dizan Vasquez, Billy Okal, and Kai O. Arras. «Inverse Reinforcement Learning algorithms and features for robot navigation in crowds: An experimental comparison». In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. Chicago, IL, USA: IEEE, Sept. 2014, pp. 1341–1346. ISBN: 978-1-4799-6934-0 978-1-4799-6931-9. DOI: 10.1109/IROS.2014.6942731. URL: http://ieeexplore.ieee.org/document/6942731/ (visited on 10/15/2024) (cit. on p. 6).

[16] Noé Pérez-Higueras, Fernando Caballero, and Luis Merino. *Learning Human-Aware Path Planning with Fully Convolutional Networks*. arXiv:1803.00429 [cs]. July 2018. URL: http://arxiv.org/abs/1803.00429 (visited on 10/15/2024) (cit. on p. 6).

[17] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. *U-Net: Convolutional Networks for Biomedical Image Segmentation*. arXiv:1505.04597 [cs]. May 2015. URL: http://arxiv.org/abs/1505.04597 (visited on 10/15/2024) (cit. on p. 8).

[18] Anthony Francis et al. *Principles and Guidelines for Evaluating Social Robot Navigation Algorithms*. arXiv:2306.16740 [cs]. Sept. 2023. URL: http://arxiv.org/abs/2306.16740 (visited on 10/15/2024) (cit. on p. 20).