

Article

A SLAM-Based Localization and Navigation System for Social Robots: The Pepper Robot Case

Tareq Alhmiedat ^{1,2,*} , Ashraf M. Marei ^{2,3}, Wassim Messoudi ⁴, Saleh Albelwi ^{2,4}, Anas Bushnag ³, Zaid Bassfar ¹ , Fady Alnajjar ⁵  and Abdelrahman Osman Elfaki ⁴

¹ Department of Information Technology, Faculty of Computers & Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia

² Artificial Intelligence and Sensing Technologies (AIST) Research Center, University of Tabuk, Tabuk 41491, Saudi Arabia

³ Department of Computer Engineering, Faculty of Computers & Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia

⁴ Department of Computer Science, Faculty of Computers & Information Technology, University of Tabuk, Tabuk 71491, Saudi Arabia

⁵ College of Information Technology, United Arab Emirates University, Al Ain 15551, United Arab Emirates

* Correspondence: t.alhmiedat@ut.edu.sa

Abstract: Robot navigation in indoor environments has become an essential task for several applications, including situations in which a mobile robot needs to travel independently to a certain location safely and using the shortest path possible. However, indoor robot navigation faces challenges, such as obstacles and a dynamic environment. This paper addresses the problem of social robot navigation in dynamic indoor environments, through developing an efficient SLAM-based localization and navigation system for service robots using the Pepper robot platform. In addition, this paper discusses the issue of developing this system in a way that allows the robot to navigate freely in complex indoor environments and efficiently interact with humans. The developed Pepper-based navigation system has been validated using the Robot Operating System (ROS), an efficient robot platform architecture, in two different indoor environments. The obtained results show an efficient navigation system with an average localization error of 0.51 m and a user acceptability level of 86.1%.

Keywords: SLAM; robot navigation; Pepper robot; social robot; Robot Operating System (ROS); map production



Citation: Alhmiedat, T.; Marei, A.M.; Messoudi, W.; Albelwi, S.; Bushnag, A.; Bassfar, Z.; Alnajjar, F.; Elfaki, A.O. A SLAM-Based Localization and Navigation System for Social Robots: The Pepper Robot Case. *Machines* **2023**, *11*, 158. <https://doi.org/10.3390/machines11020158>

Academic Editor: Shabnam Sadeghi Esfahlani

Received: 14 December 2022

Revised: 12 January 2023

Accepted: 17 January 2023

Published: 23 January 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Social robots have recently gained attention for their perceived ability to solve several challenges faced by modern society. They aim to enhance the living conditions of people who interact with them; as social robots, they have the ability to interact with humans in collaborative settings, such as homes, shopping malls, and hospitals, in which the robot may perform domestic services and healthcare tasks [1]. Therefore, social robots have interested both academics and practitioners for real-life applications.

For this, robot mapping and navigation in indoor environments are essential abilities [2–4]. In general, navigation systems consist of the following two major components: a mapping system, which produces a map of the environment, and a navigation system, which can plan and execute paths in that environment [5].

The Pepper robot is a social robot released by Softbank Robotics. It is considered as one of the most advanced social robots that was designed to allow cognitive and physical interaction with humans [6,7]. This robot has been employed in a diverse range of social applications, including healthcare, education, entertainment, and domestic settings [8–11]. To be effective, the Pepper robot needs to have a strong awareness of its surrounding environment, from recognizing individual environment components and structures to navigating safely in the area of interest, recognizing users and offering information.

One of the current uses of the Pepper robot is in the Industrial Innovation and Robotics Center (IIRC), where it performs receptionist and escort duties for IIRC visitors. It interacts with visitors through robot–human interaction system, provides a tour of the facility, and delivers information about the IIRC’s different workstations.

Pepper offers several advantages in human–robot interaction, for instance, its shape, voice, and kinematics. However, as stated in recent research [12,13], the current applications for the Pepper robot have been largely restricted due to several technical limitations, such as limited sensing capabilities, restricted API functionalities, and random, unplanned motion when turning at speeds. In addition, the NAOqi navigation package is not an open source and it offers only a limited set of navigation functionalities. Moreover, according to Reuters [14], almost 27,000 Pepper robots were produced and sold out across the world. Therefore, it is important, when investigating several methods, to develop an efficient robot navigation framework for the Pepper robot platform.

Mainly, the Pepper robot platform may be programmed using different development environments, including Choregraphe, Robot Operating System (ROS), Pepper QiSDK Android Studio, AskNAO Tablet, and AskNAO Blockly. However, the most advanced and common development environments are Choregraphe and ROS. Choregraphe offers diverse functionalities to easily develop complex robot applications, but Choregraphe also offers limited operations for the mapping, localization, and navigation functions. ROS, on the other hand, is a set of software libraries and tools that assist the developer to build robot applications, but it fails to achieve precise map production for navigation mobile robots. Therefore, it is necessary to exploit these features for both development environments to develop a reliable robot navigation system.

In order to address the above issues, this research has developed a SLAM-based navigation system, which allows the robot to self-localize and navigate itself in large halls. SLAM techniques have enabled social robots to autonomously navigate in indoor environments through building a map for unknown environments, while simultaneously keep track of the robot’s position. This paper discusses the design and development of an efficient robot navigation system in order to help the Pepper robot navigate freely in complex indoor environments. In addition, the developed system provides an efficient robot–human interaction method for introducing available devices and services to IIRC visitors. The main contributions of this paper are as follows:

1. Its review of the recently developed Pepper robot navigation systems for indoor environments.
2. Its development of a method for generating useful 2D metric maps directly from the data collected by Pepper’s onboard sensors, which overcomes the limitations in the current ROS-based map production systems.
3. Its development of efficient navigation and localization systems for navigating complex environments, using Pepper’s limited sensor suite.
4. Its presentation of a set of efficiency metrics to assess the developed systems.
5. Its assessment of user acceptability for the developed navigation system.

The remainder of this paper is organized as follows: Section 2 discusses the recently developed Pepper-based navigation systems. Section 3 discusses the proposed Pepper navigation system. Section 4 presents the experimental testbed and the results obtained from several real experiments. The obtained results are discussed in Section 5. Finally, the conclusion and future work are presented in Section 6.

2. Related Works

Robot navigation can be categorized into the following three main groups: geometric-based [15,16], semantic-based [17–19], and hybrid approaches [20,21]. This section discusses the recently developed robot navigations systems that have been developed for the Pepper social robot platform.

Only a few research works have considered the Pepper robot platform, due to the high cost of the Pepper platform, its limited onboard sensors, and complicated API. The existing

works include the research presented in [22], which involved a complete navigation framework for indoor environments with a collision avoidance function. The collected images from the robot's RGB-Depth camera were used to localize the robot within a topological graph, through adopting an image-based visual servoing (IBVS) control function. Several experiments have been performed on trails over 14 days, visiting more than 50 destination points, an experiment that achieved a success rate of approximately 80%.

The authors of [23] proposed a visual SLAM-based localization and navigation approach for service robots. The proposed system was validated using the Pepper robot platform, whose short-range LIDARs and RGB-Depth camera did not allow the robot to navigate in large environments. The system has been validated using the following two different environments: a medium-sized laboratory, and a large hall. The average success rate for the developed navigation system was around 70%.

The work presented in [13] involved the development of an automated and interactive robot system for navigating a robotics laboratory, where the developed system overcame the limitations of the robot's sensing capabilities. This work involved several contributions, including an updated motion controller, which worked with the limited sensor suite available in the Pepper robot.

The authors of [24] presented a 2D navigation system using an RGB-D camera sensor, wherein the developed system was able to extract 2D laser scans out of the 3D point cloud provided by the camera, which were later used by its mapping and localization systems. This work filled the gap between laser rangefinder (LRF) and RGB-D technology by presenting an effective mapping system based on the RGB-D camera. The authors revealed that the results obtained from employing the RGB-D camera were very similar to the maps generated by the LRF technology.

The work presented in [25] developed an efficient system with reliable navigation and localization capabilities by adding personalized interaction functions for humans based on face recognition. The obtained results offered an efficient autonomous robot navigation system using the Pepper robot.

In [26], the authors presented a real-time system for emotion-aware navigation using the Pepper robot among pedestrians, with no intensive concentration on the robot's navigation function. The developed system aimed to predict the pedestrians' emotions based on the pleasure–arousal–dominance model. In [27], the authors proposed a multi-objective navigation strategy based on machine emotions, which allowed the guide robot to predict the destinations that visitors expected to visit, based on the emotional state of the tourists.

Cuma was a robotic museum guide application based on the Pepper robot platform, as presented in [28]. Cuma accompanied visitors on a tour, providing explanations and interacting with the visitors in order to collect their feedback. The authors revealed that the provided software platform presented some critical limitations, including navigation, and thus required the integration of external tools and algorithms.

The work presented in [29] involves the development of an open-source lightweight navigation system for the Pepper robot platform. The developed navigation package overcomes several limitations that exist in the ROS development environment, it runs on the Pepper onboard PC, and offers efficient navigation capabilities.

The NavRep simulation environment has been proposed in [30] for reinforcement learning applications. NavRep aims to employ any range-based sensors for navigation purposes, and to allow anyone to reproduce state-of-the-art solutions for learning-based robot navigation approaches.

As presented above, a few Pepper robot navigation-based systems have been developed for diverse applications. Table 1 discusses the main differences among these recently developed systems based on the following parameters:

1. Navigation method: This involves the methods and algorithms that have been employed in the developed application. The employed navigation methods can be geometric-based, semantic-based, or hybrid approaches.

2. Employed sensors: different types of sensors can be integrated in the navigation task, with diverse levels of computing complexity.
3. Development environment: Pepper robot applications can be developed using either Choregraphe or ROS. Each environment has its own advantages and drawbacks.
4. System's efficiency: this involves analyzing the developed robot navigation efficiency in terms of success rate, absolute trajectory error (ATE), localization accuracy, map production, and user acceptability.

Table 1. A comparison between Pepper robot navigation systems.

Research Work	Navigation Method	Employed Sensors	Development Environment	System's Efficiency
[12]	ORB-SLAM	LiDAR, RGB-Depth, and odometry	ROS	ATE: 0.4095 m
[13]	SLAM	LiDAR	ROS/NAOQi	NA
[22]	IBVS	RGB-Depth camera	ROS	Success rate: 80%
[23]	SLAM	LiDAR and odometry	Choregraphe	Success rate: 70%
[24]	SLAM	RGB-Depth camera	ROS	Translated error: 0.115 m
[25]	SLAM	LiDAR and odometry sensors	ROS + IBM service	NA
[26]	NAOQi	LiDAR	Choregraphe	Emotion detection accuracy: 85.33%
[27]	SLAM	LiDAR	Choregraphe	NA
[28]	SLAM	LiDAR	Choregraphe	NA
[29]	Monte Carlo localization and Dijkstra	Odometry and laser scanner	C++-based environment	NA
[30]	reinforcement learning	LiDAR	NavRepSim environment	Success rate: 76%

NA: The authors did not investigate their developed system's efficiency.

As presented above, these Pepper robot navigation systems did not consider several significant evaluation parameters, including the map production accuracy, localization error, success rate, and map trajectory error. For instance, the robot's localization accuracy and map production issues were not assessed in most of the existing approaches. Therefore, this paper developed an efficient social navigation system that has been intensively assessed, where we enhanced the navigation accuracy for social robot applications through employing several reliable functions.

3. Proposed Pepper Navigation System

For implementation purposes, the ROS framework has been employed to develop the proposed robot navigation system. The ROS is not a real operating system, but its framework consists of a set of libraries and tools that have assisted developers in robot applications [31]. The ROS has been chosen due to its applicability for developing successful robotics applications. The localization and navigation functions of the Pepper robot have been implemented using the ROS framework, whereas the map production function has been developed using the Choregraphe development kit. Figure 1 depicts the main phases for this proposed navigation system.

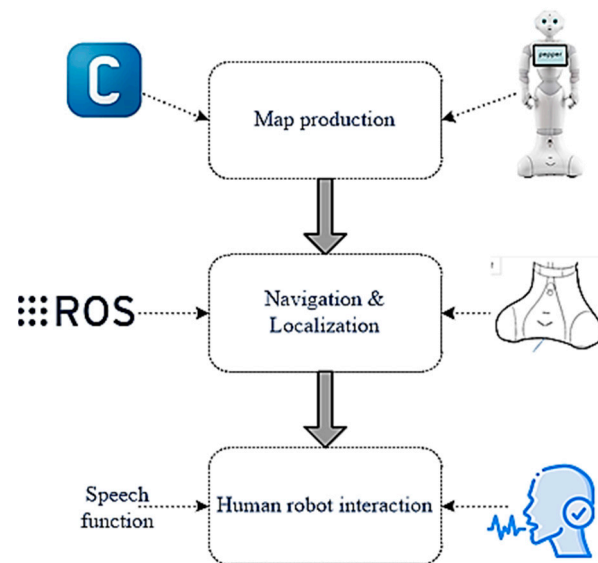


Figure 1. Main phases of the proposed navigation system.

3.1. Map Production Function

As discussed in the introduction, the maps obtained from the NAOqi_driver are low in accuracy. To combat this, there is significant demand to develop an efficient map production function. To solve the problem of low mapping accuracy, this research developed an efficient method for constructing the map area using the Choregraphe development environment.

The developed map production function works as follows: it starts by employing the explore navigation function located in the Choregraphe development environment. This function requires an initial radius value to start its exploration. In the next step, the GetMetricMap function is adopted to provide an array of pixel values in a .png format. Then, the format is converted into a program monitor (PGM) format for display purposes. Finally, the obtained map is fed into the ROS development environment in order to perform the navigation task.

From the Choregraphe development environment, this paper obtained its array of pixel values from the package `AINavigationProxy: getMetricMap` function and constructed an image for the navigation area. After this, a high-resolution map was obtained for the area of interest. This is a successful solution for the NAOqi_driver's low accuracy.

3.2. Navigation and Localization Functions

The navigation function has been developed using the ROS framework, based on the employment of the `move_base` package, in order to process Pepper's navigation task. The `move_base` package offers the implementation of a global and local planner to achieve the navigation task. It consists of several methods that allow the robot to move from one point to another using the navigation stack. An essential node in the `move_base` package is the `move_base` node, which is a major component of the navigation stack.

The navigation stack is a 2D navigation system that obtains information from sensor streams, odometry, and a goal pose and then yields safe velocity commands that are sent to the mobile base. This paper used the sensed data from the odometry and rangefinder sensors that already existed in the Pepper robot. The following equation represents the navigation function:

$$F(X) = \forall M_{x,y} [(x_1, y_1, \dots, x_n, y_n)] \supset S_i \Rightarrow R \quad (1)$$

where M denotes movement in the defined area (x, y) , S_i denotes the sensor input, and R denotes the result value.

In addition, this research employed the NAOqi_driver ROS driver for controlling the robot's motion, and then obtained the required values from Pepper's sensors. The

NAOqi_driver package is a common package for Pepper, NAO, and Romeo robot platforms, where it wraps the required parts of NAOqi API and makes them available in the ROS.

Indoor robot localization is a challenging task due to the existence of walls and obstacles in environments [32]. For localization, this paper adopted the adaptive Monte Carlo localization (AMCL) approach, which localizes the robot’s position continuously and is able to estimate the robot’s location during the navigation task. The estimation of the Pepper robot’s location is based on odometry sensors located in the Pepper robot’s base unit.

For obstacle avoidance issues, this research implemented an obstacle_avoidance node that collects sensed data from the odometry and laser sensors to determine the obstacles and allow the robot to avoid collisions, whereas the data received from the odometry sensor allows the robot to navigate itself around the obstacles.

For the purpose of controlling the path of the Pepper robot, this work developed a new node named movTopoint, which is responsible for moving the robot from one location to another. The Pepper robot needs to navigate through several predetermined points, where each point represents an object (for instance, another robot, a 3D printer, or machines).

3.3. Human–Robot Interaction Function

The proposed robot navigation system requires the Pepper robot to interact with the surrounding people in an interactive way. In addition, the robot may provide a brief speech about the station as it moves by. To accomplish this task, this research developed a speech node in the ROS (speech) that is responsible for establishing a conversation between a visitor and the Pepper robot system; it then performs speech activities at the requested positions. The speech node obtains localization information from the mov_Topoint node in order to give the correct brief speech at the correct station.

Figure 2 shows the implemented structure for Pepper’s navigation system, which consists of the move_base package, along with the implemented nodes, to achieve the navigation task. Table 2 presents the main functions that have been implemented to accomplish the social navigation task. In Figure 2, the new customized nodes are presented in blue (speech, moveTo_point, and obstacle_avoidance).

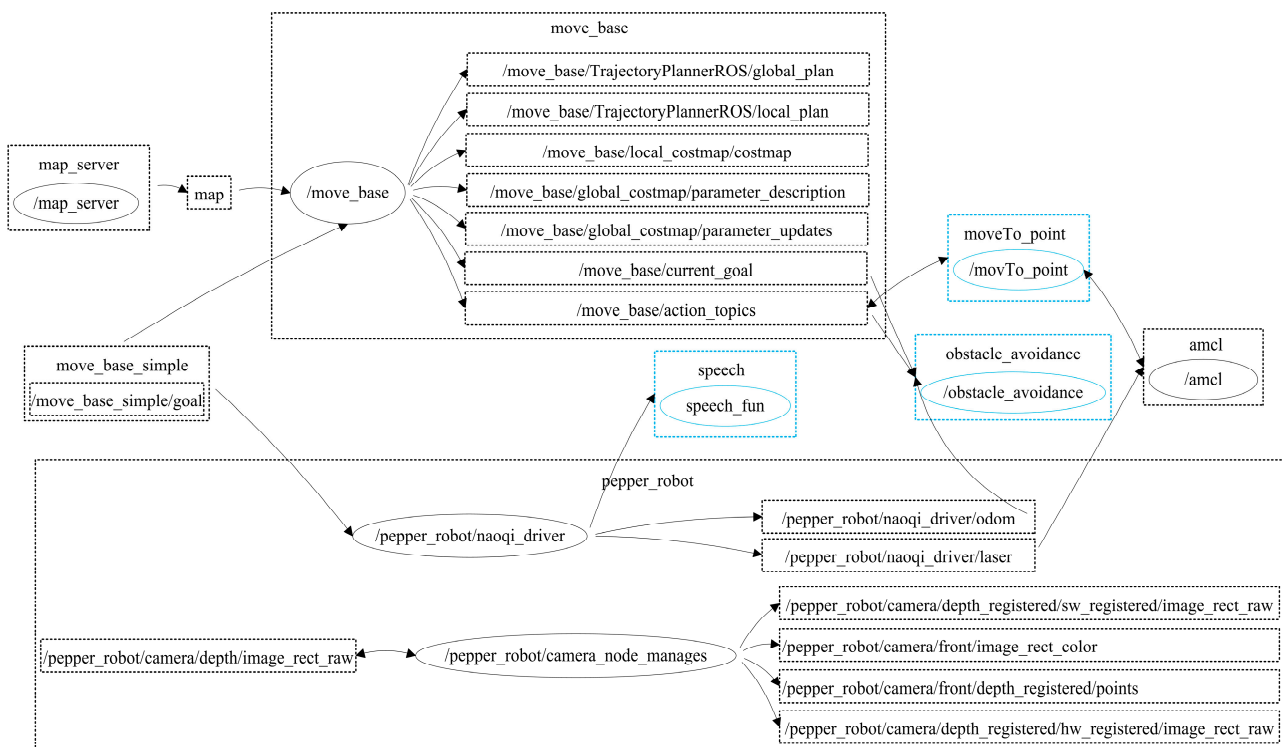


Figure 2. The architecture of the developed ROS-based navigation framework.

Table 2. A summary of the developed navigation functions.

Function	Input	Output
Obstacle avoidance	Rangefinder sensor data	New route through move_base
Move-to-point	Area ID from the list of stations	The area's coordinates are sent to the move_base
Speech	User interface function	Short speech

4. Experiments and Results

4.1. Experimental Setup

For validation purposes, this paper considered the following two different indoor environments at the University of Tabuk: a study room in the IIRC, as shown in Figure 3 and referred to as Area 1, and second, the FabLab area in the IIRC, as presented in Figure 4 and referred to as Area 2. Area 1 is a study area with a dimension of $14.1 \times 3.92 \text{ m}^2$ that consists of a number of desks and chairs, whereas Area 2 is a lab area with a dimension of $20.4 \times 7.6 \text{ m}^2$ that includes desks, chairs, 3D printers, robots, electronic equipment, and work tables. The selected environments were different in size, furniture, and visual appearance complexity.

The Pepper robot platform has been chosen to test the developed navigation system. Pepper is a child-sized robot platform, as depicted in Figure 5. Its mobile robot platform can move using three omnidirectional wheels; it has 20 degrees of movement from the 17 joints on its body. Pepper's body is made from white plastic and is equipped with a tablet, capacitive sensors, and loudspeakers to assist in its interactions. In addition, Pepper has four microphones, a sonar sensor, a three-dimensional sensor, two RGB cameras, infrared sensors, laser sensing modules, and bumper sensors in order to assist the robot with identifying people and objects around its body. The Pepper robot has the ability to interact with people through speech, and it has LED lights in its eyes and ears to aid in its communication of emotions.

**Figure 3.** Side view of Area 1 (study area).



Figure 4. Side view of Area 2 (IIRC Fablab).



Figure 5. The Pepper robot platform, located in the IIRC.

For the development environment, this research employed both Choregraphe and the ROS. The Choregraphe environment has been utilized to perform the map production function, whereas the ROS framework has been adopted to implement the navigation tasks, including localization, obstacle avoidance, and robot–human interaction. ROS framework has been employed in several robotics applications, including robot navigation and localization.

Map production is a requirement in mobile robot navigation, as the Pepper robot needs to localize and navigate itself in the area of interest. The map production function was carried out autonomously, in which the robot traveled in the area of interest and built a 2D map area. The map production task was accomplished in the following two ways: through the ROS and Choregraphe. In the ROS architecture, the map was built using

only the rangefinder sensors, which obtained an inaccurate map. The combined array of rangefinder sensors offered a max range of 1.5 m, thus reducing mapping accuracy.

This research initially attempted to employ the ROS gmapping framework. However, the generated maps were extremely corrupted, even at short distances. Therefore, to overcome this issue, this paper then employed the Choregraphe development environment to generate useful maps at high resolution. The Choregraphe-based generated map achieved better production accuracy, as the generated map was obtained from both the rangefinder and the odometer sensors employed in the Pepper platform. The navigation task was accomplished using the global and location planner package in the ROS development environment.

Figure 6 shows the actual structure for Area 1 (study area), whereas Figure 7 presents the actual structure for Area 2 (IIRC lab), along with the pre-defined station points that the Pepper robot needs to visit and provide a brief description.

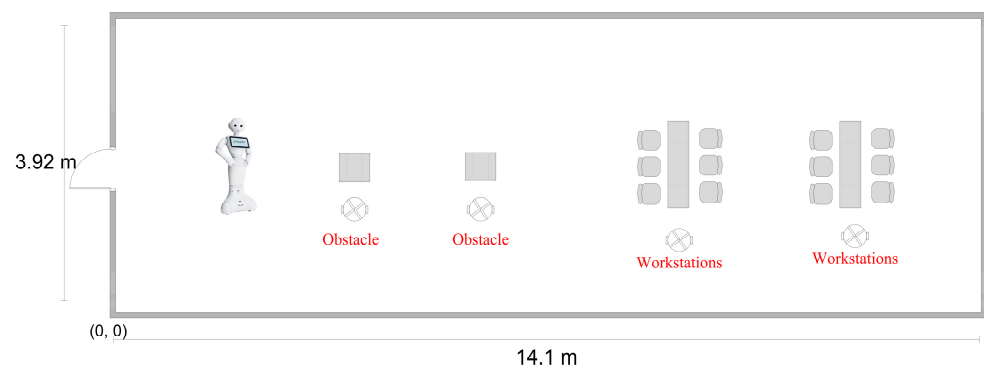


Figure 6. The actual structure of Area 1 (study area).

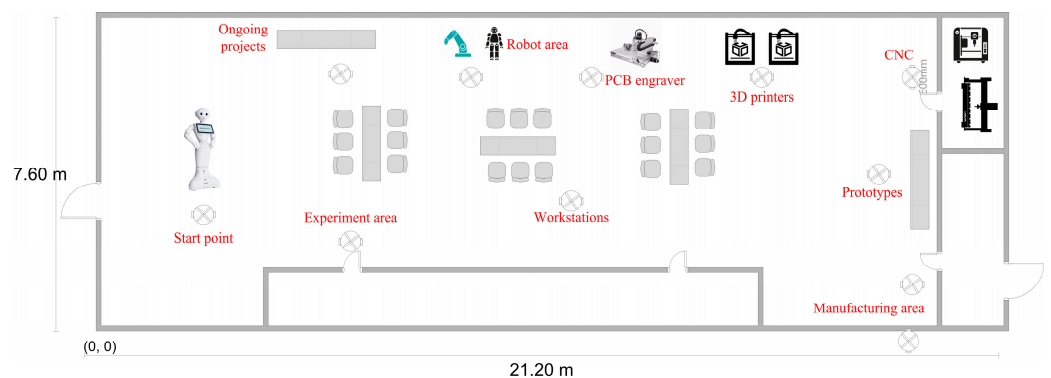


Figure 7. The actual structure of Area 2 (IIRC lab), along with predefined station points.

4.2. Results

This section presents the efficiency metrics for social robot navigation, and discusses the results obtained from several real experiments conducted in two different indoor environments. The authors of [33,34] reviewed the common metrics used to evaluate socially away robot navigation systems, including navigation efficiency, success rate, and sociability. Therefore, in this paper, we extended the evaluation metrics in order to precisely assess the efficiency of social robot navigation systems. The adopted evaluation metrics are as follows:

- Map production task: this assesses the accuracy of the produced map for the area of interest.
- Robot localization error: this estimates the difference between the robot's actual position and its estimated position.

- Absolute trajectory error (ATE): this shows the average trajectory error for the robot when traveling in the area of interest.
- Success rate: This measures the robot's ability to reach its goal. In addition, the success rate involves the number of collisions and timeouts.
- Robot path: this shows Pepper's safe path to navigate from one reference point to another in the area of interest.
- User acceptability: this refers to the overall acceptability of the social robot navigation system by its users.

For validation purposes, this research developed a demonstration in which the Pepper robot would provide autonomous tours in the IIRC laboratory (Area 2). Pepper was designed to move between 10 different stations within the environment.

The map construction function was performed in both Areas 1 and 2. The obtained map dimensions for Area 1 were $13.0 \times 3.54 \text{ m}^2$, with a total error size of 9.25 m^2 , and an average map production error of 16.7%, whereas the obtained map dimensions for Area 2 were $20.52 \times 6.4 \text{ m}^2$, with a total error size of 23.72 m^2 and an average map production error 15.1%. As noted above, the obtained localization error ratios for the two areas (Areas 1 and 2) are almost the same with a slight variance, with an average of 15.9%.

The developed navigation system allows the robot to perform the map production task first. Figure 8 presents the map area obtained for Area 1, whereas Figure 9 shows the obtained map area for Area 2 (IIRC lab). However, the average map production accuracy was around 84.1% for the two environment testbeds. Hence, the maps generated using the production function had higher mapping accuracy than the map produced using the ROS package.



Figure 8. The obtained map of Area 1.



Figure 9. The obtained map of Area 2.

The maps for Areas 1 and 2 have been generated using Choregraphe, with a resolution of 0.05 pixel/meter. Figure 10 shows the production map accuracy for Areas 1 and 2 using the Choregraphe- and ROS-based function, respectively. As noticed, the developed Choregraphe-based function achieved the best map production accuracy for both scenarios (Areas 1 and 2).

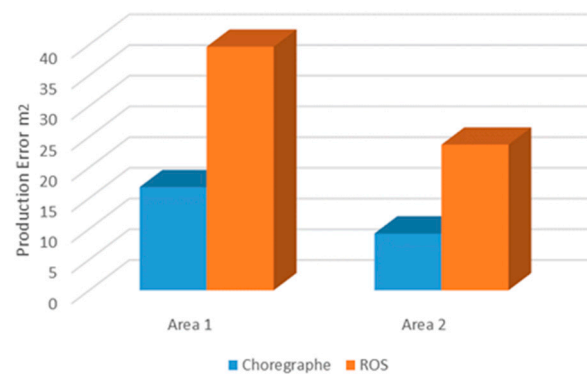


Figure 10. The average map production error for the two areas using Choregraphe and ROS.

The robot's localization accuracy was a critical part of the development of this navigation system, since the Pepper robot needs to travel to different, predefined points in order to perform its tour task. Therefore, this section analyzes the average localization error when the Pepper robot system travels in the navigation areas.

The localization accuracy was estimated along 10 reference points, where each reference point represents a station point that the robot needs to stop at and offer a brief description. The localization accuracy for each reference point was estimated by calculating the difference between the robot's estimated location ($x_e - y_e$) and the robot's real location ($x_r - y_r$), according to the following formula:

$$Loc_{Acc} = \sqrt{(x_e - x_r)^2 + (y_e - y_r)^2} \quad (2)$$

In both experiment testbeds, there were 10 different stations, as presented earlier in Figure 6. Pepper's location was estimated at each of these 10 different positions, where, for each estimation, the differences between the robot's actual location and the estimated location, as obtained from the developed localization function, were calculated. The obtained results showed average localization errors of 0.43 and 0.60 m for Area 1 and 2, respectively. Figure 11 presents the localization error for Area 1 through several reference points, whereas Figure 12 shows the localization error for Area 2 through several reference points.

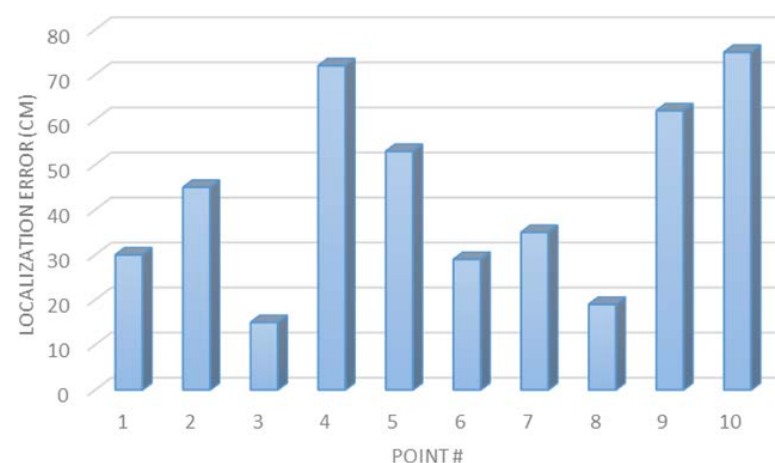


Figure 11. The localization error for the robot system in Area 1 for 10 different positions.

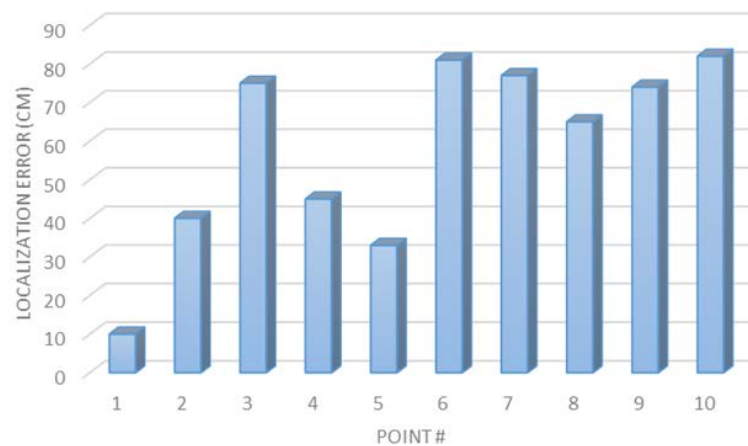


Figure 12. The localization error for the robot system in Area 2 for 10 different positions.

Absolute trajectory error (ATE) is a metric that calculates the root mean square error (RMSE) between the estimated trajectory point m_{e_i} and the actual trajectory point m_{a_i} [35], where the ATE is defined as follows:

$$ATE = \left(\frac{1}{N} \sum_i^N |m_{e_i} - m_{a_i}|^2 \right)^{\frac{1}{2}} \quad (3)$$

where N refers to the total points in the trajectory. Table 3 presents the ATE for two different environments (Areas 1 and 2). During the experiments conducted in Areas 1 and 2, the Pepper robot must move smoothly, and preferably sideways, to estimate the robot's locations. The map production function refers to the employment of rangefinder sensors and the odometer attached to the Pepper robot. The trajectory error was estimated for 10 different points, as discussed in the previous section. Table 3 shows the ATE results for Areas 1 and 2.

Table 3. Absolute trajectory error (ATE) in centimeters for Areas 1 and 2 and axis (x and y).

Experiment Testbed	ATE (x Coordinate)	ATE (y Coordinate)
Area 1	54	31
Area 2	66	51

The robot's success rate measures Pepper's ability to reach a particular point of interest within a reasonable period and in a safe manner. Usually, the success rate is highly dependent upon the environmental characteristics. In our case, the success rate estimates the ability of the Pepper robot to reach its goal without a collision. The collision rate is the rate that the Pepper robot terminates its navigation task due to collisions with dynamic objects and humans. The timeout metric, on the other hand, refers to the rate that the Pepper robot is unable to reach the destination point within the time limit. Pepper's success rate has been evaluated for the following two experimental testbeds: testbed 1 and testbed 2. Table 4 shows the success rate, collision rate, and the timeout for the two experiment testbeds.

Table 4. Quantitative results including the success rate and collision rate for two areas.

Experiment Testbed	Success Rate	Collision Rate	Timeout
Area 1	0.92	0.01	0.001
Area 2	0.89	0.18	0.115

As noticed in Table 4, the success rate for Area 1 is better than Area 2, since Area 1 includes less obstacles and consists of four stations points to travel through. The same result is observed for the collision rate and timeout, where the collision rate increases when

more dynamic objects are located in the navigation area, as with Area 2. In addition, the time needed to reach the destination points in Area 2 is longer than the time needed for Area 1, due to the existence of dynamic objects, which may cause the robot to fail to reach the destination point within the required time.

The Pepper robot’s path was also assessed for two experiment testbeds. Estimating the Pepper robot’s path is an essential task to analyze the success of the developed robot navigation system. Figure 13 presents the robot’s path for Area 1, whereas Figure 14 shows the robot’s path for Area 2.

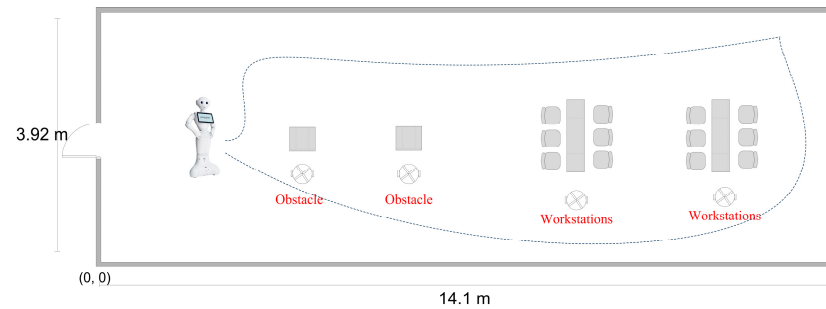


Figure 13. Pepper’s path in Area 1.

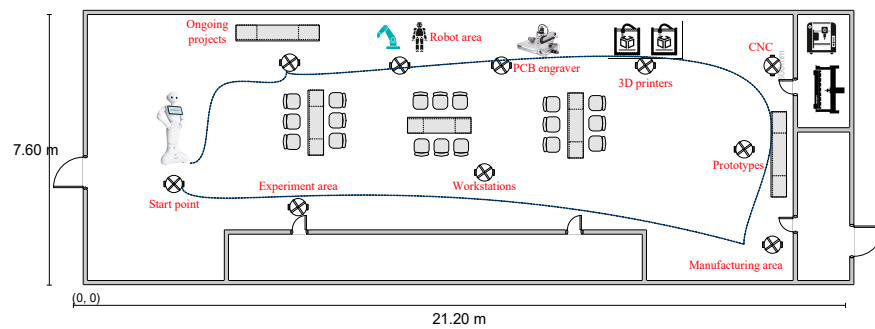


Figure 14. Pepper’s path in Area 2.

According to [27], social robot acceptability can be assessed based on the following several factors: the role assigned to the robot, the robot’s social capabilities, and the robot’s appearance. Therefore, the work presented in this paper focused on the above three factors. For instance, the main function of the developed system is to interact with, and navigate visitors to, the main stations in the IIRC lab. Figure 15 presents the users’ acceptance of the offered functions by the developed Pepper robot system, with an average acceptability result of 87.2%. Most of the visitors showed a high level of acceptance when working with the robot system as a new tool for guiding visitors through the IIRC.

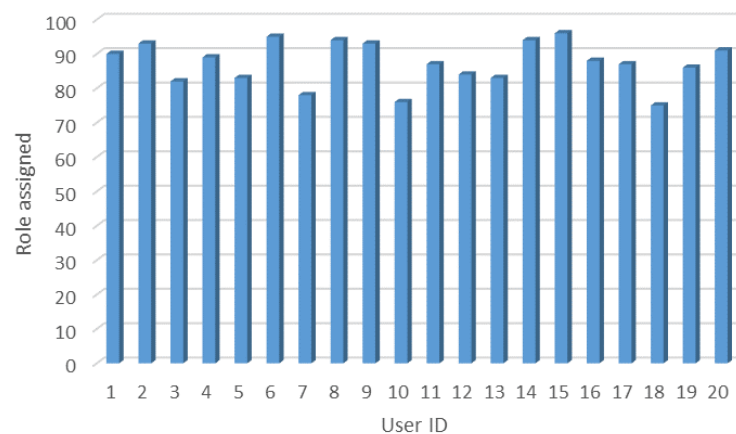


Figure 15. Assessment of the acceptance of the role assigned to the robot.

The Pepper robot is considered to be a friendly, social interaction robot, and the employed system has been developed to interact with visitors in a cooperative way and perform different emotions. The authors of [36] even classified the Pepper robot as human-like; therefore, Pepper succeeds at the task of robot appearance, as Pepper is human-like. As noted in Figure 16, the average acceptance of the robot for its social interaction function was almost 86.1%. Pepper was able to provide an illustration of the devices and equipment placed in the IIRC lab, in an interactive way. A high ratio of the visitors found Pepper to be an efficient device in terms of social capabilities.

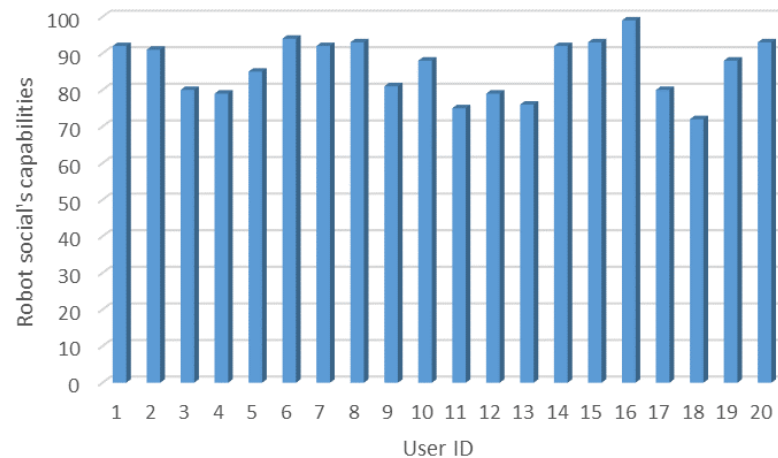


Figure 16. Assessment of the robot social's capabilities.

In addition, the Pepper robot's appearance was assessed from the users' perspectives. As presented earlier in Figure 5, Pepper has a plastic shell for skin, colorful eyes, and a mouth, which increases the robot's ability to display human-like emotions [37]. Figure 17 presents the acceptability of the robot's appearance to visitors of the IIRC lab. As noted, most of the visitors found Pepper to be effective in terms of appearance, with an average acceptance rate of 90.1%. They found that the Pepper robot has a good shape and is human-like.

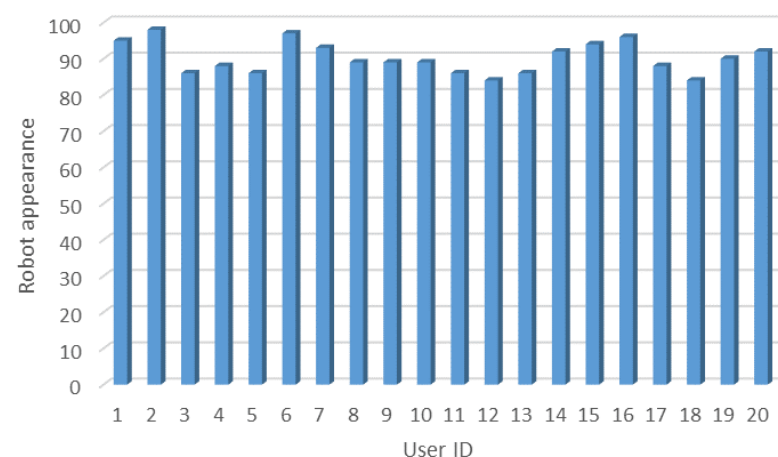


Figure 17. Assessment of the robot's appearance from the user's perspective.

5. Discussion

The Pepper robot platform can be developed using the Choregraphe and ROS platforms. The Choregraphe development environment offers limited base functionalities for the navigation and mapping tasks, whereas the ROS fails to obtain an accurate map for the navigation area [12,13], hence producing an unreliable robot navigation system for indoor

environments. The authors of [38] further attest to this, arguing that Pepper is not suitable for autonomous exploration, navigation, or SLAM tasks.

For the purpose of robot navigation, this paper implemented three different functions (nodes) in order to overcome the aforementioned limitations and perform the tasks of robot navigation in the IIRC area, with a focus on the following aspects: obstacle avoidance, move-to-point, and speech nodes.

Vision-based navigation systems [22–24] are efficient in terms of map production efficacy and navigation accuracy. However, the Pepper robot platform has limited memory size and processor speed, both of which slow down the processing of vision-based navigation systems, where there is a high demand for these in order to perform the processing required for robot vision images.

The work presented in [22,23,27] used the Choregraphe development environment, which restricts the performance of several functions, including mapping and navigation. The proposed navigation system developed in this research overcomes several limitations in the existing navigation systems [12,13], through the employment of a novel map production method using Choregraphe, which overcomes the limitations of the ROS production map function. In addition, the proposed navigation system has been implemented and evaluated in two different environments (a medium-sized area and a large lab), using a 2D laser scanner and odometry sensors. The quantitative experimental results proved the effectiveness of the proposed system, which was able to generate 2D maps, where the quality was estimated based on the difference between the real and the generated maps.

According to [39], social assistive robots refer to robots that are designed to aid people in a way that focuses on social interactions, such as guiding, speaking, reminding, observing, and entertaining. One of the main domains of social robots is the guiding or companion robot, where the robot needs to support elderly people, particularly those who are living alone. Unlike the works presented in [12,23,24,38], this paper investigates the visitors' acceptability for the robot application, by adopting a reliable robot–human interaction system. According to the obtained results, the developed navigation system offers a reasonable acceptance level.

Moreover, the developed navigation system achieves a low localization error when navigating indoor environments, with an average localization error of 0.51 m. The developed navigation system offers efficient localization accuracy, meaning that the Pepper robot arrives at the required station point with the minimum localization error.

6. Conclusions

The Pepper robot has been employed in diverse types of social applications, in which robot navigation is essential for such applications. However, Pepper's navigation task is challenging with the available development environments. This paper presents the development of an autonomous robot navigation system for the Pepper robot platform. This work has been framed in the context of offering interactive tours in an open-plan laboratory, focusing on the following two main goals: enhancing the Pepper robot's autonomy and improving its ability to interact with IIRC visitors. The former goal has been achieved through the development of an efficient navigation system based on the ROS and Choregraphe, where a new efficient map production function has been developed. The latter has been accomplished by implementing an efficient speech recognition system. The developed navigation system has been validated in two different environments, in which it achieved efficient map production, localization, and navigation. In addition, this paper offers a set of evaluation metrics for assessing the efficiency of any social robot navigation system. Future work will aim to develop a semantic navigation system for the Pepper robot platform by employing an RGB-D camera to further enhance navigation accuracy.

Author Contributions: Both A.O.E. and S.A. surveyed the recent developed systems. T.A. developed the robot navigation system and performed several experiments. A.M.M., W.M., F.A., A.B. and Z.B. validated and documented the obtained results. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Deanship of Scientific Research (DSR) at the University of Tabuk, Tabuk, Saudi Arabia, under grant no. 1441-105.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would also like to acknowledge the financial support for this work from the Deanship of Scientific Research (DSR) at the University of Tabuk, Tabuk, Saudi Arabia, under grant no. 1441-105.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Krägeloh, C.U.; Bharatharaj, J.; Kutty, S.K.S.; Nirmala, P.R.; Huang, L. Questionnaires to Measure Acceptability of Social Robots: A Critical Review. *Robotics* **2019**, *8*, 88. [CrossRef]
2. Kim, P.; Chen, J.; Kim, J.; Cho, Y.K. SLAM-driven intelligent autonomous mobile robot navigation for construction applications. In *Workshop of the European Group for Intelligent Computing in Engineering*; Springer: Cham, Switzerland, 2018; pp. 254–269. [CrossRef]
3. Ross, R.; Hoque, R. Augmenting GPS with Geolocated Fiducials to Improve Accuracy for Mobile Robot Applications. *Appl. Sci.* **2019**, *10*, 146. [CrossRef]
4. Alhmiedat, T.A.; Abutaleb, A.; Samara, G. A Prototype Navigation System for Guiding Blind People Indoors using NXT Mindstorms. *Int. J. Online Biomed. Eng. (ijOE)* **2013**, *9*, 52. [CrossRef]
5. Alamri, S.; Alshehri, S.; Alshehri, W.; Alamri, H.; Alaklabi, A.; Alhmiedat, T. Autonomous maze solving robotics: Algorithms and systems. *Int. J. Mech. Eng. Robot. Res.* **2021**, *10*, 12. [CrossRef]
6. Efstratiou, R.; Karatsioras, C.; Papadopoulou, M.; Papadopoulou, C.; Lytridis, C.; Bazinas, C.; Papakostas, G.A.; Kaburlasos, V.G. Teaching Daily Life Skills in Autism Spectrum Disorder (ASD) Interventions Using the Social Robot Pepper. In *Proceedings of the International Conference on Robotics in Education (RiE)*; Springer: Cham, Switzerland, 2020; pp. 86–97. [CrossRef]
7. De Jong, M.; Zhang, K.; Roth, A.M.; Rhodes, T.; Schmucker, R.; Zhou, C.; Ferreira, S.; Cartucho, J.; Veloso, M. Towards a robust interactive and learning social robot. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, Stockholm, Sweden, 10–15 July 2018; pp. 883–891.
8. Schrum, M.; Park, C.H.; Howard, A. Humanoid therapy robot for encouraging exercise in dementia patients. In *Proceedings of the 2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, Daegu, Republic of Korea, 11–14 March 2019; IEEE: Manhattan, NY, USA, 2019; pp. 564–565.
9. Corallo, F.; Maresca, G.; Formica, C.; Bonanno, L.; Bramanti, A.; Parasporo, N.; Giambò, F.M.; De Cola, M.C.; Buono, V.L. Humanoid Robot Use in Cognitive Rehabilitation of Patients with Severe Brain Injury: A Pilot Study. *J. Clin. Med.* **2022**, *11*, 2940. [CrossRef]
10. Ziouzos, D.; Rammos, D.; Bratitsis, T.; Dasygenis, M. Utilizing Educational Robotics for Environmental Empathy Cultivation in Primary Schools. *Electronics* **2021**, *10*, 2389. [CrossRef]
11. Getson, C.; Nejat, G. Socially Assistive Robots Helping Older Adults through the Pandemic and Life after COVID-19. *Robotics* **2021**, *10*, 106. [CrossRef]
12. Gómez, C.; Mattamala, M.; Resink, T.; Ruiz-Del-Solar, J. Visual SLAM-Based Localization and Navigation for Service Robots: The Pepper Case. In *Robot World Cup*; Springer: Cham, Switzerland, 2018; pp. 32–44. [CrossRef]
13. Suddrey, G.; Jacobson, A.; Ward, B. Enabling a pepper robot to provide automated and interactive tours of a robotics laboratory. *arXiv* **2018**, arXiv:1804.03288.
14. Nussey, S. EXCLUSIVE SoftBank Shrinks Robotics Business, Stops Pepper Production—Sources. *Reuters*, 29 June 2021. Available online: <https://www.reuters.com/technology/exclusive-softbank-shrinks-robotics-business-stops-pepper-production-sources-2021-06-28/> (accessed on 16 January 2023).
15. Ravankar, A.; Ravankar, A.A.; Kobayashi, Y.; Hoshino, Y.; Peng, C.-C. Path Smoothing Techniques in Robot Navigation: State-of-the-Art, Current and Future Challenges. *Sensors* **2018**, *18*, 3170. [CrossRef]
16. Gul, F.; Rahiman, W.; Alhady, S.S.N.; Chen, K. A comprehensive study for robot navigation techniques. *Cogent Eng.* **2019**, *6*, 1632046. [CrossRef]
17. Alenzi, Z.; Alenzi, E.; Alqasir, M.; Alruwaili, M.; Alhmiedat, T.; Alia, O.M. A Semantic Classification Approach for Indoor Robot Navigation. *Electronics* **2022**, *11*, 2063. [CrossRef]
18. Crespo, J.; Castillo, J.C.; Mozos, O.M.; Barber, R. Semantic Information for Robot Navigation: A Survey. *Appl. Sci.* **2020**, *10*, 497. [CrossRef]
19. Joo, S.-H.; Manzoor, S.; Rocha, Y.G.; Bae, S.-H.; Lee, K.-H.; Kuc, T.-Y.; Kim, M. Autonomous Navigation Framework for Intelligent Robots Based on a Semantic Environment Modeling. *Appl. Sci.* **2020**, *10*, 3219. [CrossRef]
20. Shi, Y.; Zhang, W.; Yao, Z.; Li, M.; Liang, Z.; Cao, Z.; Zhang, H.; Huang, Q. Design of a Hybrid Indoor Location System Based on Multi-Sensor Fusion for Robot Navigation. *Sensors* **2018**, *18*, 3581. [CrossRef]

21. Moezzi, R.; Krcmarik, D.; Hlava, J.; Cýrus, J. Hybrid SLAM modelling of autonomous robot with augmented reality device. *Mater. Today Proc.* **2020**, *32*, 103–107. [[CrossRef](#)]
22. Bista, S.R.; Ward, B.; Corke, P. Image-Based Indoor Topological Navigation with Collision Avoidance for Resource-Constrained Mobile Robots. *J. Intell. Robot. Syst.* **2021**, *102*, 1–24. [[CrossRef](#)]
23. Silva, J.R.; Simão, M.; Mendes, N.; Neto, P. Navigation and obstacle avoidance: A case study using Pepper robot. In Proceedings of the IECON 2019-45th Annual Conference of the IEEE Industrial Electronics Society, Lisbon, Portugal, 14–17 October 2019; IEEE: Manhattan, NY, USA, 2019; Volume 1, pp. 5263–5268.
24. Nardi, F.; Lázaro, M.T.; Iocchi, L.; Grisetti, G. Generation of Laser-Quality 2D Navigation Maps from RGB-D Sensors. In *Robot World Cup*; Springer: Cham, Switzerland, 2019; pp. 238–250. [[CrossRef](#)]
25. Perera, V.; Pereira, T.; Connell, J.; Veloso, M. Setting up pepper for autonomous navigation and personalized interaction with users. *arXiv* **2017**, arXiv:1704.04797.
26. Bera, A.; Randhavane, T.; Prinja, R.; Kapsaskis, K.; Wang, A.; Gray, K.; Manocha, D. The emotionally intelligent robot: Improving social navigation in crowded environments. *arXiv* **2019**, arXiv:1903.03217.
27. Chen, D.; Ge, Y. Multi-Objective Navigation Strategy for Guide Robot Based on Machine Emotion. *Electronics* **2022**, *11*, 2482. [[CrossRef](#)]
28. Allegra, D.; Alessandro, F.; Santoro, C.; Stanco, F. Experiences in Using the Pepper Robotic Platform for Museum Assistance Applications. In Proceedings of the 2018 25th IEEE International Conference on Image Processing (ICIP), Athens, Greece, 7–10 October 2018; IEEE: Manhattan, NY, USA, 2018; pp. 1033–1037.
29. Lázaro, M.T.; Grisetti, G.; Iocchi, L.; Fentanes, J.P.; Hanheide, M. A Lightweight Navigation System for Mobile Robots. In Proceedings of the Iberian Robotics Conference; Springer: Cham, Switzerland, 2018; pp. 295–306. [[CrossRef](#)]
30. Dugas, D.; Nieto, J.; Siegwart, R.; Chung, J.J. NavRep: Unsupervised Representations for Reinforcement Learning of Robot Navigation in Dynamic Human Environments. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; IEEE: Manhattan, NY, USA, 2021; pp. 7829–7835.
31. Quigley, M.; Conley, K.; Gerkey, B.; Faust, J.; Foote, T.; Leibs, J.; Wheeler, R.; Ng, A.Y. ROS: An open-source Robot Operating System. ICRA Workshop on Open Source Software; IEEE: Kobe, Japan, 2009; Volume 3, p. 5.
32. Alhmiedat, T.; Aborokbah, M. Social Distance Monitoring Approach Using Wearable Smart Tags. *Electronics* **2021**, *10*, 2435. [[CrossRef](#)]
33. Gao, Y.; Huang, C.-M. Evaluation of Socially-Aware Robot Navigation. *Front. Robot. AI* **2022**, *8*, 420. [[CrossRef](#)] [[PubMed](#)]
34. Nishimura, M.; Yonetani, R. L2B: Learning to Balance the Safety-Efficiency Trade-off in Interactive Crowd-aware Robot Navigation. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NE, USA, 25–29 October 2020; IEEE: Manhattan, NY, USA, 2020; pp. 11004–11010.
35. Sturm, J.; Engelhard, N.; Endres, F.; Burgard, W.; Cremers, D. A benchmark for the evaluation of RGB-D SLAM systems. In Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, Vilamoura-Algarve, Portugal, 7–12 October 2012; IEEE: Manhattan, NY, USA, 2012; pp. 573–580.
36. Chanseau, A.; Dautenhahn, K.; Walters, M.L.; Koay, K.L.; Lakatos, G.; Salem, M. Does the Appearance of a Robot Influence People's Perception of Task Criticality? In Proceedings of the 2018 27th IEEE International Symposium on Robot and Human Interactive Communication (RO-MAN), Nanjing, China, 27–31 August 2018; IEEE: Manhattan, NY, USA, 2018; pp. 1057–1062.
37. Onyeulo, E.B.; Gandhi, V. What Makes a Social Robot Good at Interacting with Humans? *Information* **2020**, *11*, 43. [[CrossRef](#)]
38. Groot, R. Autonomous Exploration and Navigation with the Pepper Robot. Master's Thesis, Utrecht University, Utrecht, The Netherlands, 2018.
39. Ghiță, A.; Gavril, A.F.; Nan, M.; Hoteit, B.; Awada, I.A.; Sorici, A.; Mocanu, I.G.; Florea, A.M. The AMIRO Social Robotics Framework: Deployment and Evaluation on the Pepper Robot. *Sensors* **2020**, *20*, 7271. [[CrossRef](#)] [[PubMed](#)]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.