Nav2CAN: Achieving Context Aware Navigation in ROS2 Using Nav2 and RGB-D sensing

Schwörer, Tristan; Schmidt, Jonathan Eichild; Chrysostomou, Dimitrios

Published in:
IST 2023 - IEEE International Conference on Imaging Systems and Techniques, Proceedings

DOI (link to publication from Publisher):
10.1109/IST59124.2023.10355731

Publication date:
2023

Document Version
Accepted author manuscript, peer reviewed version

Nav2CAN: Achieving Context Aware Navigation in ROS2 Using Nav2 and RGB-D sensing

Tristan Schwörer
Dept. of Materials and Production
Aalborg University
Aalborg, Denmark
tristans@mp.aau.dk

Jonathan Eichild Schmidt
Dept. of Architecture, Design and Media Technology
Aalborg University
Aalborg, Denmark
jesc@create.aau.dk

Dimitrios Chrysostomou
Dept. of Materials and Production
Aalborg University
Aalborg, Denmark
dimi@mp.aau.dk

***Abstract*—This paper presents a real-time human interaction detection system using RGB-D cameras to enable context-aware navigation for mobile robots. The system employs a convolutional neural network (CNN) architecture optimized for efficient inference on embedded GPUs. Using a keypoint detection based human detector on RGB-D images, interactions are localized in the 3D scene using object detection of humans. The detected human interaction zones are integrated into the robot's navigation costmaps to modify planned paths accounting for social spaces. The system is validated through simulated and real-world tests showing reliable interaction detection at over 10 Hz. The modular system, called Nav2CAN, can be added to mobile robots operating in *ROS2* (Robot Operating System 2) and achieve easy integration and compatibility with other packages. By combining deep learning-based perception with semantic navigation costmaps, socially-aware robot navigation in human environments is achieved.**

***Index Terms*—Context Aware Navigation, Human-Robot Interaction, Mobile Robots, ROS2, Nav2, Proxemics**

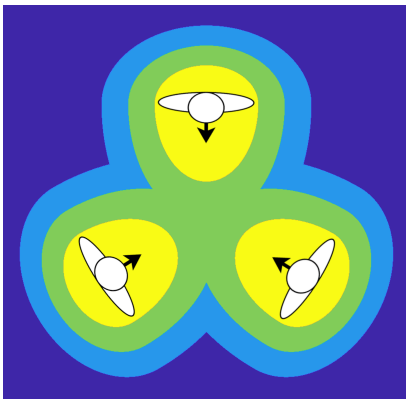

Fig. 1. Group of people represented by individual egg-shaped proxemic zones. Yellow represents the intimate zone, green the personal zone, light blue the social zone and dark blue indicates the public zone. For visual purposes, not to scale.

I. Introduction

As mobile robots become more prevalent in human environments, developing navigation systems that understand and react to social contexts is critical. Context aware navigation is extending the obstacle avoidance task of conventional navigation with an understanding of the robot's surrounding. In the case of social and service robotics, these surroundings are often filled with humans, which creates a necessity to understand their actions in order to navigate in a comfortable and safe manner.

One approach is to leverage proxemics - the study of human use of interpersonal space as introduced by Edward T. Hall [1]: *the interrelated observations and theories of humans use of space as a specialised elaboration of culture* and defined as four zones occupying the space surrounding humans. However, the definition of these proxemic zones have been expanded, resulting in, among others, more precise definitions of the zone parameters and the representations of actions performed by humans in the scene such as moving [2], interacting [3] and forming groups [4] as shown in Fig. 1.

Some assumptions must be made, however, for a given robotic platform to implement context aware navigation. Therefore, determining which social rules the robot must abide to is essential to guide the implementation of such a robotic system. For this paper the definition of the social interactions between humans are defined by the humans standing in close vicinity of one another - within the social zones of the proxemic zone. This means that the robot must be aware of these interactions when planning navigation. In doing so, the humans will be more likely in accepting the working robot and thereby improve the collaboration between human and robot [5]–[9]. However, this awareness results in a need for a detection-capability for the chosen interactions and represent them such that the robot can avoid violating the associated social rules.

This work is not the first that seeks to explore context or social awareness [10]–[13]. However, while previous works have explored proxemics-based navigation, they often lack modularity and compatibility with the latest robotics software frameworks rendering them incompatible for use in realistic working environments. This paper presents Nav2CAN - a modular system built on ROS2 and the popular Nav2 [14], [15] navigation stack to enable context aware navigation. Nav2CAN detects people and their interactions to generate costmap plugins that respect social norms. This allows seamless integration of social awareness into existing navigation pipelines.

The key contributions are: 1) flexible architecture for context detection and costmap generation using ROS2 nodes

and plugins, 2) a deep learning-based method for detecting social interactions, 3) validation in simulations and real-world module tests showing improved awareness and comparison to prior methods. In addition, by providing Nav2CAN as open source repository to the research community, we strive to enable further research and implementations of socially-aware navigation for mobile robots [1].

The remainder of the paper is organized as follows. Section III describes the methods used for Nav2CAN as well as the considerations needed to achieve context aware navigation as a plugin for Nav2. Section IV shows the results of the implemented system and how it compares to other human aware navigation systems. The paper is concluded in section VI.

II. Related Work

Proxemics are often leveraged in order to represent individual humans in a costmap based navigation stack, such as Nav2 as shown by Clavero et al. [10] where they alter the proxemic zones in the case of a wanted collaboration. Here, the wanted collaboration depends on the user and is therefore hard to generalize [16]. Marques et al. [11] proposed a proxemics-based approach for handling activity in the cost map of the robot. This includes the detection of a human interacting with a given object and increasing the affordance space. To understand affordance spaces, the relation between objects are necessary [17], [18].

For a robot to act on affordance spaces, determining the action and location of humans are important to achieve comfortable navigation, as described by Garg et al. [19]. In this context, navigation is deemed successful when an interaction that is perceived positively by the humans involved. Law et al. [6] show how important the movement of a robot, whether humanoid, animal-shaped, or any other shape, influences the trust of said robot. It is found that movement that follows expected patterns and social standards improves the trust and comfort around the robot which is important to ensure that humans want to use the robot [20].

Mavrogiannis et al. [7] describes the importance of understanding humans as more than just dynamic obstacles in an environment to achieve more than just collision avoidance. Assuming that the human will seek to avoid collision will improve the trajectory of the robot, resulting in less oscillatory movement. Babel et al. [21] takes another step in understanding human behaviour by analysing potential conflicts between humans and robots when working in the same environment. The authors argue that a robot that always yields can become inefficient in the long run and therefore must be able to interrupt the user. Different methods have been tested for an assertive robot which shows that being polite and using knowledge from psychology results in higher trust from the user [12].

For understanding the humans for navigation, it is important to determine which type of activity must be recognised. Li et al. [22] describes the necessity to understand the social relationships between humans in a scene to understand the interaction. This means that understanding the relationship will enable a system, such as a robot, to predict the actions of interacting humans. Kostevalis et al. [23] used a skeleton-based approach to understand the activity of the human by combining the actions of the skeleton with the associated object in a spatio-temporal network. Sadeghian et al. [13] uses social and physical cues to understand the potential paths of humans walking in a scene. Vemula et al. [24] developed Social attention to predict the trajectories of humans walking among each other while Li et al. employed natural language processing frameworks [25], [26] and task dialogue systems [27] to enable more natural interaction of the mobile robots with the human operators.

[1]https://github.com/Nav2CAN/main

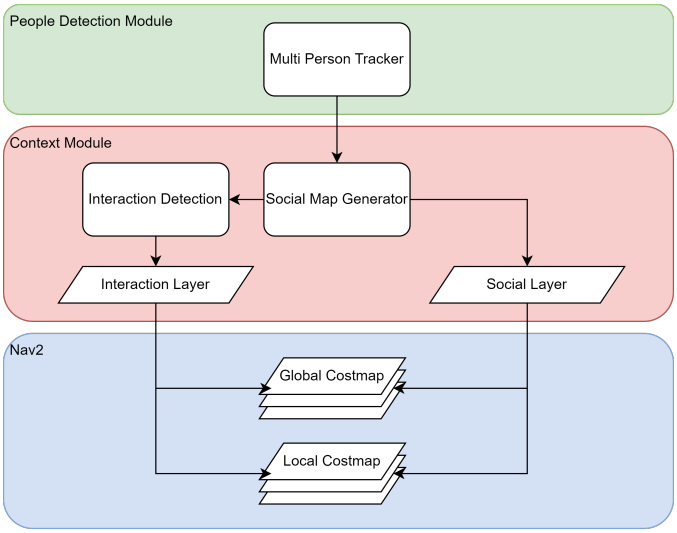

Fig. 2. Nav2CAN modular system overview. Green represents the people detection module with standardised output. Red represents the context module that is transforming the information of the people in the environment into the form of costmap plugins. Blue represents a Nav2 navigation of any configuration.

III. Methods

The system overview of Nav2CAN, is shown in Fig. 2. Nav2CAN is separated into three modules the *People Detection Module* (green), the *Context Module* (red) and the *Nav2* navigation stack (blue).

As shown in Fig. 2 the Context module requires the input of the people in the surrounding environment of the robot. In our system, a Kalman filter [28] based multi person tracker has been deployed that detects people in front of the robot using two Intel RealSense D435 [29] which provide the option of aligning depth and RGB images. This enables the use of detection in RGB to estimate distance from depth.

In order to allow the usage of different people detectors the message type for the communication between the detector and the context module needs to be standardised. Here inspiration has been taken from the package *people_msgs* from ROS1 and its message types type *people_msgs/People* and *people_msgs/Person* [30] have been re-implemented for ROS2

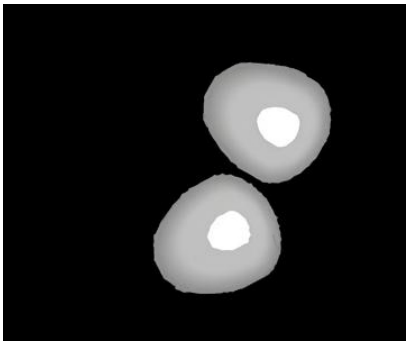

Fig. 3. An example of the generated social map. Here two people are present and their proxemics have been generated, showing that the people are facing away from each other.

such that they can be used for Nav2CAN. This message is also required to include the *tf_frame* that the measurement is relative to and a *tf_tree* between that *tf_frame* and the *map tf_frame* should be supplied.

As our system strives towards being modular and flexible, the output of the context module needs to be defined such that it can be used for numerous different applications. Here the costmap based navigation framework Nav2 is used since it offers a wide range of different planning and controlling algorithms that are all based on costmaps for a wide range of robots. This allows to define the output of the context module to be costmap layer plugins.

A. Context module

As mentioned in Section I, the main contribution of this paper is the context module called Nav2CAN. It includes the necessary nodes to detect the context of a scene and represent it in a way that is interpretable by conventional costmap based planners available for Nav2.

The input for the interaction detection is designed to be a grey scale image such that proxemics are included in the input. Therefore, a node is required that generates this output from the poses of all people in the environment. An example of the generated output is shown in Fig. 3

The resulting value distribution is then thresholded according to (1), resulting in the cost distribution. Where the Gaussian function reflects a Gaussian distribution at the origin with the variance of θ_{side} and $value$ is the one calculated according to (2) [31].

$$cost = \begin{cases} maxcost, & \text{if value>Gaussian(0.5)} \\ maxcost * Gaussian(1.0), & \text{if value>Gaussian(1.0)} \\ value * maxcost, & \text{if value>Gaussian(1.5)} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$$value(x, y, theta) = e^{-(A(x-x_i)^2+2B(x-x_i)(y-y_i)+C(y-y_i)^2)} \tag{2}$$

Where:

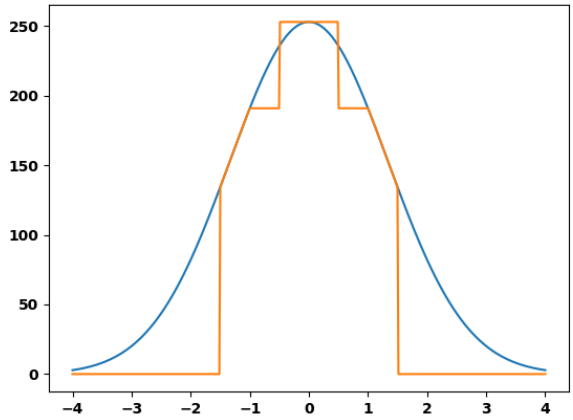

Fig. 4. An example of the thresholded cost distribution used for calculating the social map. Here only one slice of the distribution is calculated. The line in blue represents the regular Gaussian while the orange function represents the thresholded values. The horizontal axis contains the distance from the center in meters while the vertical axis contains the cost value.

- $A = \frac{\cos(\theta)^2}{2*\theta^2} + \frac{\sin(\theta)^2}{2*\theta^2_{side}}$
- $B = \frac{\sin(2\theta)^2}{4*\theta^2} - \frac{\sin(2\theta)^2}{4*\theta^2_{side}}$
- $C = \frac{\sin(\theta)^2}{2*\theta^2} + \frac{\cos(\theta)^2}{2*\theta^2_{side}}$
- θ_{side} is the variance to left and right
- θ is variance to the front or back depending on the pixel coordinate

An example of the thresholded Gaussian is shown in Fig. 4.

The reason for the threshold applied to the cost instead of the smooth asymmetric Gaussian is to define values for the different proxemic zones instead of smoothing their transition. For the experiments for this paper, the intimate zone gets a unreachable high value of 253 so the robot can never advance into that zone. The personal zone, however, is divided into a fixed cost section and a decaying section. The reasoning for this is that the advancing of the robot into the personal zone should be discouraged by increasing the cost at the edge of the zone. As the robot might be required to enter the personal zone of a person in order to interact with them, the cost is not further increased within that zone, such that robot can position itself anywhere within it. This is meant to force the robot to take the shortest possible path through the personal zone to its goal. To prevent the people flooding the entire costmap with low cost values as the Gaussian decays, the distribution is clipped off at the edge of the personal zone such that the robot is free in its movement.

The resulting map from the individual people represents the people at the time of recording relative to the current position of the robot, using the *tf_frames*. This social map, is centred at the robots position but is not rotated; instead it is aligned to the *map* frame. As a result, interactions in the map are not moving drastically when the robot rotates in order to avoid the people in the environment. With precise localisation, this

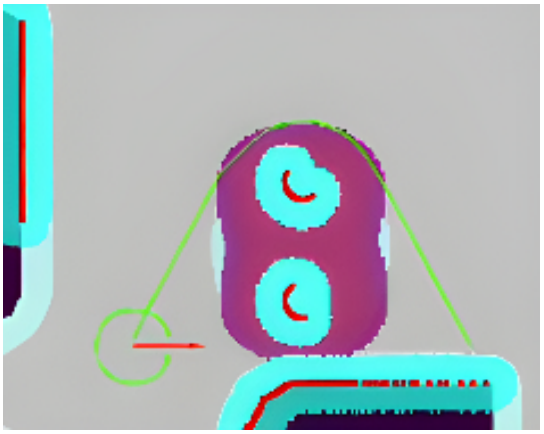

Fig. 5. Path of Nav2CAN taken around an interaction of two people that face each other in green. The purple area shows the proxemic zones of the two people, who are shown as laser-scan in red and turquoise half circles. Underneath the proxemics is the interaction zone in turquoise. The colors represent different values in the costmap.

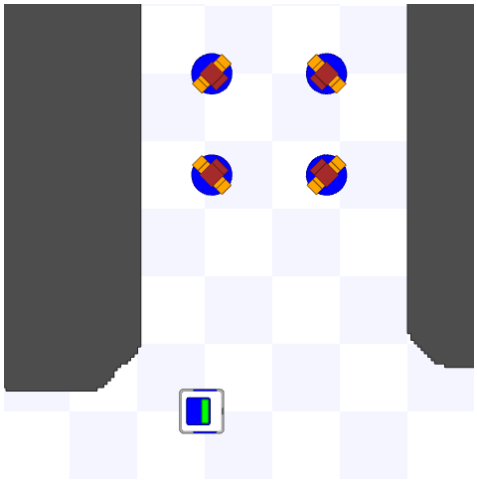

Fig. 6. An example of the testing environment from simulation. Here four people are standing in a social interaction where the robot needs to move from on side to the other. The squares of the map are $1m \times 1m$.

simplifies the combination with the local costmap as the local costmap is aligned to the *odom* frame.

As the proxemic zone is a grid of cost values, it is computationally expensive to calculate. Thus the cost distribution is calculated during initialisation of the node and then rotated and copied onto the social map according to the pose of each detected person. To detect interactions from the above mentioned social map, a detector is used to determine which areas of the map contain interactions. The interaction detection in this work is based on the YOLOv7 architecture [32].

The trained network is constructed as a ROS2 node that subscribes to the generated social map and detects interactions within the map. The detections are then re-published as a message of type *BoundingBox*, which described the location and size of the interaction.

The resulting costmaps from the social map and the interaction detection can then be generated to use with Nav2. For the social map, a map is already generated for the interaction detector and the image can therefore be converted to a costmap grid by representing the pixel values as cost values and thereby publishing these values to the Nav2 stack for either the local or global costmap or alternatively for both.

Similar to the social costmap layer, the interaction layer is also implemented as a *costmap_2d* layer and therefore a plugin that can be added to any Nav2 based navigation stack. The task of this layer is to draw elliptical cost distributions based on the bounding boxes detected by the interaction pipeline. The shape of the elliptical zone is an approximation of the P space for interaction zones, described by Kendon [2]. As it is unknown whether this layer is implemented in the global, local or both costmaps, the coordinates of the center have to be transformed into the global *tf_frame* of the affected costmap and the current time. This avoids a swerving cost distribution that might largely hinder path planning and navigation. From the received bounding box, the ellipse is drawn into the map layer using the specified value in the Nav2 parameter setup. An example of the resulting costmaps can be seen in Fig. 5.

IV. Results

The planner of the navigation stack used in our work is the default planner *nav2_navfn_planner/NavfnPlanner*. This planner is then accompanied with the smoother *nav2_smoother::SimpleSmoother*, in order to remove edges caused by the grid based path finding techniques and to improve path feasibility. The path following is handled by the controller plugin *dwb_core::DWBLocalPlanner*. In the experiments performed for this paper, the interaction layer is added to the global costmap such that movement through interactions can be discouraged by using an ellipse for the interaction with a high cost. The social layer is added to the local costmap such that the controller can steer the robot in accordance to the defined proxemic constraints.

Nav2CAN is tested against the CoHAN Planner [33] to determine how the context aware navigation fares against an established planner for human aware navigation. A simulation environment is available for CoHAN and it has therefore been utilised for comparing the capabilities of the two systems. The resulting simulation environments contain humans in various groups of varying sizes to test the performance of the interaction detector and the generated social map for path planning. The two planners are compared in their distance to people as well of how they disturb social proxemics zones. As the CoHAN planner is not developed for interaction detection the results serve as a indication of how much a system, such as Nav2CAN, adds values to social mobile robots. An example of the test environment is depicted in Fig. 6.

For social interactions between two and four people, CoHAN shows a tendency to move between people given enough space while Nav2CAN avoids this entirely and plans a route outside of the interaction zone. This means that CoHAN will navigate the robot within $0.72\mathrm{m}$ in-front of the person.

TABLE I
SHORTEST DISTANCE (IN METERS) BETWEEN PERSON CENTRE AND ROBOT CENTRE DURING EACH SCENARIO.

Navigation system	CoHAN	Nav2CAN
Scenario		
2 people - personal zone	0.89	0.94
2 people - social zone	1.00	3.50
3 people in a group	0.76	2.88
4 people in a group	0.72	3.00

As there are multiple options for detecting people in an environment, the results presented in this paper only describe a subset of these possibilities. With the simple detection system developed for testing, a RGB-D camera was used in a surveillance setup where the location of people was used for Nav2CAN. This results in the system being deployable to a ROS2 network. The tests of the system show that people located in the camera can be cast into the associated costmaps and thereby enable context aware navigation.

It is also important for the navigation system of a context aware mobile robot to operate in real-time. Nav2CAN proves to perform faster than 10hz by having an average calculation time of less than 89ms with an unoptimized setup on a Nvidia Jetson AGX Orin developer kit [34]. This includes the detector developed for testing. The detection systems were developed for running on the edge with the Orin mounted to the mobile robot. This meant the developed interaction detector, using the Pytorch [35] framework, could run locally on the robot making the system able to run offline, given sufficient computational power.

V. DISCUSSION & FUTURE WORK

The experiments demonstrate Nav2CAN's ability to avoid interrupting detected social interactions compared to baseline methods, as shown by the metrics in Table I. Qualitative observations also indicate smoother, more human-aware navigation trajectories around people versus simpler proximity-based reactive planning. However, limitations exist in relying solely on proxemics and current interaction approaches. As humans move through spaces interacting with one another and objects, their positions and relationships are constantly changing. More complex social norms will require additional contextual understanding as for example, detecting office conversations vs. transient hallway discussions may warrant different costmap representations.

Future work will focus on increasing the contextual awareness beyond interactions and spatial zones. Skeletal tracking and pose analysis could provide richer understanding of human movements and activities. Incorporating affordances based on objects, locations, and layouts will allow responding appropriately across different cultural and interaction contexts. Graph-based representations could also capture the relationships between people, objects, and locations to enable more sophisticated reasoning.

An immediate progression of this work will enable the overall system to be validated on our real-world robotic platforms, known as Little Helpers [36] (latest iteration shown in Fig. 7), while navigating through a real manufacturing lab. This progression might highlight new challenges or limitations due to occlusions of humans in the scene. A major challenge will be to achieve long-term autonomy as it will require larger capacity in memory and recurrent inference to gain acquaintance knowledge of patterns, habits, and behaviours in a space. Alternative sensing modalities such as speech and dialogue systems may also help perceive social cues while accurate intention and trajectory prediction could be achieved by exploring probabilistic and generative models of human behaviour.

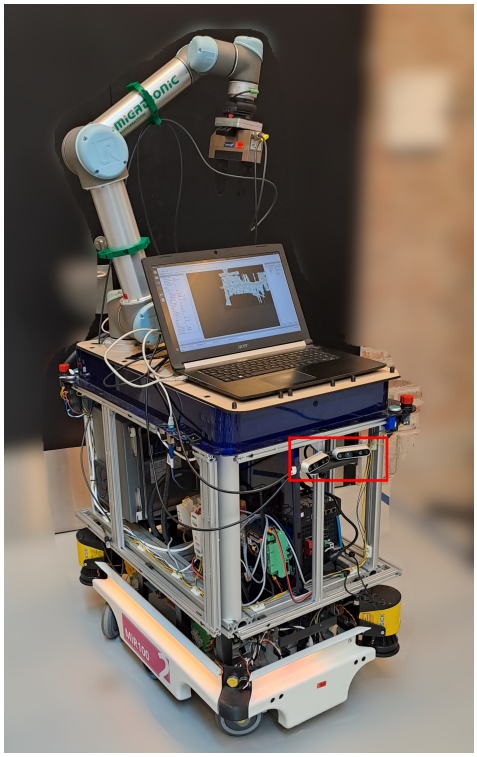

Fig. 7. Little Helper [36], based on a MIR100 [37] mobile platform with the dual Intel RealSense D435 RGB-D sensor setup marked in red.

VI. CONCLUSION

This paper presents Nav2CAN, a modular system for achieving context-aware navigation in ROS2 using the Nav2 stack. The system incorporates detected human interactions into costmap layers to shape planned paths that respect social spaces. The experiments illustrate improved social awareness compared to proximity-based methods, while also revealing limitations of the current approach due to the dynamic nature of the interactions with humans.

However, substantial future research remains to achieve fully human-aware assistive robots. This will require holistic integration of perception, inference, intention prediction, path and behavioural planning and interaction across multiple timescales. As robots are increasingly deployed in domains like healthcare, retail, and hospitality, they must balance task

planning with social intelligence. Social intelligence for robots is a complex challenge spanning technology, ethics, design, policy, and cross-cultural perspectives. The examined method, Nav2CAN, represents an initial step towards realizing this long-term vision of seamless navigation and collaboration between humans and social and service robots in diverse human populated environments.

REFERENCES

[1] E. Hall, "The hidden dimension. an anthropologist examines man's use of space in public and private. new york: Anchor books; doubleday & company, inc," 1969.

[2] A. Kendon, "Spacing and orientation in co-present interaction," *Development of Multimodal Interfaces: Active Listening and Synchrony: Second COST 2102 International Training School, Dublin, Ireland, March 23-27, 2009, Revised Selected Papers*, pp. 1–15, 2010.

[3] M. Gérin-Lajoie, C. L. Richards, J. Fung, and B. J. McFadyen, "Characteristics of personal space during obstacle circumvention in physical and virtual environments," *Gait & posture*, vol. 27, no. 2, pp. 239–247, 2008.

[4] M.-L. Barnaud, N. Morgado, R. Palluel-Germain, J. Diard, and A. Spalanzani, "Proxemics models for human-aware navigation in robotics: Grounding interaction and personal space models in experimental data from psychology," in *Proceedings of the 3rd IROS'2014 workshop "Assistance and Service Robotics in a Human Environment"*, 2014.

[5] N. Savela, T. Turja, and A. Oksanen, "Social acceptance of robots in different occupational fields: a systematic literature review," *International Journal of Social Robotics*, vol. 10, no. 4, pp. 493–502, 2018.

[6] T. Law, J. de Leeuw, and J. H. Long, "How movements of a non-humanoid robot affect emotional perceptions and trust," *International Journal of Social Robotics*, vol. 13, pp. 1967–1978, 2021.

[7] C. Mavrogiannis, P. Alves-Oliveira, W. Thomason, and R. A. Knepper, "Social momentum: Design and evaluation of a framework for socially competent robot navigation," *ACM Transactions on Human-Robot Interaction (THRI)*, vol. 11, no. 2, pp. 1–37, 2022.

[8] A. Chatzimichali, R. Harrison, and D. Chrysostomou, "Toward privacy-sensitive human–robot interaction: Privacy terms and human–data interaction in the personal robot era," *Paladyn, Journal of Behavioral Robotics*, vol. 12, no. 1, pp. 160–174, 2020.

[9] C. Jost, B. Le Pévédic, T. Belpaeme, C. Bethel, D. Chrysostomou, N. Crook, M. Grandgeorge, and N. Mirnig, *Human-Robot Interaction*. Springer, 2020.

[10] J. G. Clavero, F. M. Rico, F. J. Rodríguez-Lera, J. M. G. Hernández, and V. M. Olivera, "Defining adaptive proxemic zones for activity-aware navigation," in *Workshop of Physical Agents*. Springer, Cham, 2020, pp. 3–17.

[11] F. Marques, D. Gonçalves, J. Barata, and P. Santana, "Human-aware navigation for autonomous mobile robots for intra-factory logistics," in *Symbiotic Interaction: 6th International Workshop, Symbiotic 2017, Eindhoven, The Netherlands, December 18–19, 2017, Revised Selected Papers 6*. Springer, 2018, pp. 79–85.

[12] F. Babel, J. M. Kraus, and M. Baumann, "Development and testing of psychological conflict resolution strategies for assertive robots to resolve human–robot goal conflict," *Frontiers in Robotics and AI*, vol. 7, 2021. [Online]. Available: https://www.frontiersin.org/articles/10.3389/frobt.2020.591448

[13] A. Sadeghian, V. Kosaraju, A. Sadeghian, N. Hirose, H. Rezatofighi, and S. Savarese, "Sophie: An attentive gan for predicting paths compliant to social and physical constraints," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 1349–1358.

[14] S. Macenski, F. Martin, R. White, and J. Ginés Clavero, "The marathon 2: A navigation system," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020.

[15] S. Macenski, T. Moore, D. Lu, and M. F. A. Merzlyakov, "From the desks of ros maintainers: A survey of modern and capable mobile robotics algorithms in the robot operating system 2," *Robotics and Autonomous Systems*, 2023.

[16] K. Youssef, S. Said, S. Alkork, and T. Beyrouthy, "A survey on recent advances in social robotics," *Robotics*, vol. 11, no. 4, p. 75, 2022.

[17] V. S. Chen, P. Varma, R. Krishna, M. Bernstein, C. Re, and L. Fei-Fei, "Scene graph prediction with limited labels," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019.

[18] J. Gu, H. Zhao, Z. Lin, S. Li, J. Cai, and M. Ling, "Scene graph generation with external knowledge and image reconstruction," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[19] S. Garg, N. Sünderhauf, F. Dayoub, D. Morrison, A. Cosgun, G. Carneiro, Q. Wu, T.-J. Chin, I. Reid, S. Gould *et al.*, "Semantics for robotic mapping, perception and interaction: A survey," *Foundations and Trends® in Robotics*, vol. 8, no. 1–2, pp. 1–224, 2020.

[20] M. Adamik, A. P. Madsen, and M. Rehm, "Explainability in collaborative robotics: The effect of informing the user on task performance and trust," in *2022 31st IEEE International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2022, pp. 1252–1257.

[21] F. Babel, A. Vogt, P. Hock, J. Kraus, F. Angerer, T. Seufert, and M. Baumann, "Step aside! vr-based evaluation of adaptive robot conflict resolution strategies for domestic service robots," *International Journal of Social Robotics*, vol. 14, no. 5, pp. 1239–1260, 2022.

[22] J. Li, Y. Wong, Q. Zhao, and M. S. Kankanhalli, "Visual social relationship recognition," *International Journal of Computer Vision*, vol. 128, pp. 1750–1764, 2020.

[23] I. Kostavelis, M. Vasileiadis, E. Skartados, A. Kargakos, D. Giakoumis, C.-S. Bouganis, and D. Tzovaras, "Understanding of human behavior with a robotic agent through daily activity analysis," *International Journal of Social Robotics*, vol. 11, pp. 437–462, 2019.

[24] A. Vemula, K. Muelling, and J. Oh, "Social attention: Modeling attention in human crowds," in *2018 IEEE international Conference on Robotics and Automation (ICRA)*. IEEE, 2018, pp. 4601–4607.

[25] C. Li, J. Park, H. Kim, and D. Chrysostomou, "How can i help you? an intelligent virtual assistant for industrial robots," in *Companion of the 2021 ACM/IEEE International Conference on Human-Robot Interaction*, 2021, pp. 220–224.

[26] C. Li, A. K. Hansen, D. Chrysostomou, S. Bøgh, and O. Madsen, "Bringing a natural language-enabled virtual assistant to industrial mobile robots for learning, training and assistance of manufacturing tasks," in *2022 IEEE/SICE International Symposium on System Integration (SII)*. IEEE, 2022, pp. 238–243.

[27] C. Li, X. Zhang, D. Chrysostomou, and H. Yang, "Tod4ir: A humanised task-oriented dialogue system for industrial robots," *IEEE Access*, vol. 10, pp. 91 631–91 649, 2022.

[28] R. E. Kalman, "A new approach to linear filtering and prediction problems," 1960.

[29] Intel, "Intel realsense d400 series product family," https://www.intel.com/content/dam/support/us/en/documents/emerging-technologies/intel-realsense-technology/Intel-RealSense-D400-Series-Datasheet.pdf, 2019, accessed on 04.04.2023.

[30] D. Lazewatsky and D. V. Lu, "ros people_msgs," 2021. [Online]. Available: https://github.com/wg-perception/people/tree/melodic/people_msgs

[31] J. Gines Clavero, F. Martín Rico, F. J. Rodríguez-Lera, J. M. Guerrero Hernández, and V. Matellán Olivera, "Impact of decision-making system in social navigation," *Multimedia Tools and Applications*, vol. 81, no. 3, pp. 3459–3481, 2022.

[32] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," 2022. [Online]. Available: https://arxiv.org/abs/2207.02696

[33] P. T. Singamaneni, A. Favier, and R. Alami, "Human-aware navigation planner for diverse human-robot ineraction contexts," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[34] "Nvidia jetson orin." [Online]. Available: https://www.nvidia.com/en-us/autonomous-machines/embedded-systems/jetson-orin/

[35] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," 2019.

[36] R. E. Andersen, E. B. Hansen, D. Cerny, S. Madsen, B. Pulendralingam, S. Bøgh, and D. Chrysostomou, "Integration of a skill-based collaborative mobile robot in a smart cyber-physical environment," *Procedia Manufacturing*, vol. 11, pp. 114–123, 2017.

[37] M. M. industrial Robots, "Mir100 datasheet 2016," https://www.i-botics.de/wp-content/uploads/2016/08/MiR100-DE-05-2016.pdf, 2016, accessed on 04.04.2023.