SEAN: Social Environment for Autonomous Navigation

N. Tsoi	M. Hussein	J. Espinoza	X. Ruiz	M. Vázquez
Yale University	Rutgers University	Yale University	Yale University	Yale University
nathan.tsoi@yale.edu				

ABSTRACT

Social navigation research is performed on a variety of robotic platforms, scenarios, and environments. Making comparisons between navigation algorithms is challenging because of the effort involved in building these systems and the diversity of platforms used by the community; nonetheless, evaluation is critical to understanding progress in the field. In a step towards reproducible evaluation of social navigation algorithms, we propose the Social Environment for Autonomous Navigation (SEAN). SEAN is a high visual fidelity, open source, and extensible social navigation simulation platform which includes a toolkit for evaluation of navigation algorithms. We demonstrate SEAN and its evaluation toolkit in two environments with dynamic pedestrians and using two different robots.

1 INTRODUCTION

Simulation is useful along the whole development cycle of robotic systems including data collection, features development, testing, and deployment [14, 19]. Simulation is key for the verification of safety-critical systems and is particularly relevant for companies that make robots for mainstream audiences [16].

Driven by the gaming industry and demand for autonomous vehicles, the robotics community has recently experienced a rapid increase in the quality and features available in simulation tools. These advancements led to simulation environments for self-driving vehicles [1, 4] and aerial vehicles [6, 13, 20]. Crowd simulations have improved as well [2, 3, 22], although often independently of simulation environments for mobile robots, including environments that build on game engines [8, 10], or state-of-the-art rendering like Gibson[26] or ISAAC.[1] This disconnect has led to a gap in high fidelity simulation environments for evaluating social robot navigation in pedestrian settings, e.g., service robots that need to operate nearby people and subject to social conventions. Our work, depicted in Figure 1, is a step towards filling this gap.

We propose SEAN, a Social Environment for Autonomous Navigation, as an extensible and open source simulation platform. SEAN includes animated human characters useful for studying human-robot social interactions in the context of navigation. As other recent simulators [8, 10, 13, 20], SEAN leverages modern graphics and physics modeling tools from the gaming industry, providing a flexible development environment in comparison to more traditional robotics simulators like Gazebo [9]. We provide two ready-to-use scenes with components that allow social agents to navigate according to standard pedestrian models. We provide integration with the Robot Operating System (ROS), which allows for compatibility with existing navigation software stacks. An important contribution of this work is a toolkit for repeated execution of navigation tasks and logging of navigation metrics. We hope SEAN facilitates developing, testing and evaluating social navigation algorithms in the future.

---

[1]https://developer.nvidia.com/isaac-sim

Figure 1: SEAN's rendering a world (left) and view of the scene from the robot's perspective (right). The outdoor city scene (top) and the lab scene (bottom) include dynamic pedestrians for studying social robot navigation.

2 SIMULATION PLATFORM

SEAN is a collection of tools in the Unity 3D game engine[2] and the Robot Operating System (ROS) [18] that allows for control of a mobile robot in a dynamic, simulated human environment. Unity implements the NVIDIA PhysX physics engine, which has been found to provide promising results for robot simulation [10]. Communication between ROS and Unity is implemented as a set of scripts executed as part of the Unity scripting run-time model and implemented via the ROS# library.[3] SEAN's architecture balances between a) ease of integration with navigation systems (or robot teleoperation) via ROS, b) high visual fidelity for creating immersive environments and enable vision-based navigation methods, and c) a cross-platform ecosystem that supports iterative development. SEAN works in Windows 10 and Ubuntu 18.04 with ROS Melodic.

The key tenets of our approach are usability and flexibility. While these often seem at odds, we seek these goals by providing a set of scenes, robots, and evaluation metrics within the platform to enable users to use the system with minimal preliminary work. Additionally, we maintain an open source repository and supporting documentation to allow the community to improve our social navigation environment.[4]. Our contributions are an effort to begin to explore the challenging problem of fairly and reproducibly benchmarking algorithms for human-robot social interactions.

Scenes: A scene is a 3D environment in which a robot operates. With our initial release, we provide a high-fidelity model of a

---

[2]https://unity.com/
[3]https://github.com/siemens/ros-sharp
[4]https://sean.interactive-machines.com/

Table 1: Sample Jackal (J) robot and Warthog (W) robot results. Controlled via the ROS Navigation Stack or teleoperated as indicated by *. $\mu \pm \sigma$ over 10 episodes.

Scene	Robot	Elapsed (sec.)	Complete	Final Dist (m)	Ped. Dist (m)	Collisions
Lab	J	24.51 ± 19.36	60%	2.26 ± 2.92m	1.54 ± 1.76m	7.1 ± 9.4
Lab	J *	21.6 ± 28.08	88%	1.14 ± 1.99	0.92 ± 1.16	4.63 ± 5.83
City	J	37.09 ± 13.74	29%	9.54 ± 8.94	0.64 ± 0.42	20 ± 30.83
City	J *	38.54 ± 29.5	80%	4.59 ± 11.87	1.06 ± 0.67	3.1 ± 7.58
City	W *	31.7 ± 20.94	100%	0.48 ± 0.01	2.27 ± 1.08	0 ± 0

lab environment and a larger outdoor city scene (Fig. 1). Because humans play a key part in the study of robot navigation in these environments, for each scene we have created reasonable start and goal positions for human agents to navigate. To this end, SEAN uses a combination of crowd flow prediction [21] and Unity's built-in path planning algorithm. The system is parameterized such that we can easily deploy an appropriate number of agents given the size and context of the scene. We can also vary the density of pedestrians across experiments in a repeatable manner. SEAN's online documentation explains how to create and modify scenes.

Robots and Sensors: SEAN provides 2 robot models ready to run: a medium size Clearpath Jackal, which is suitable for indoor and flat outdoor environments; and a Warthog with 254mm of ground clearance. The Warthog is more suitable for outdoor environments due to its bigger size (Fig. 1). Because neither robot comes equipped with standard sensors, we outfitted them with a simulated Velodyne VLP-16,[5] a LIDAR scanner, and a simulated RGB camera.

Evaluation Toolkit: SEAN's toolkit for evaluating social navigation algorithms centers on the Trial Runner, which enables repeatable and automatic execution of navigation tasks. The Trial Runner performs a *trial* by executing a collection of point-to-point navigation *episodes*. Each episode begins with the Trial runner configuring the scene, actors, and robot positions. Pedestrians are assigned goal positions and a ROS navigation goal is used to indicate the desired final pose for the robot. As the robot navigates, the Trial Runner records relevant metrics. It starts a new episode once the robot has moved to a sufficiently close location to the destination or the episode times out. While the initial conditions for each episode are random by default, they are recorded at the beginning of an episode. This allows to replay the episode for fair comparisons of navigation methods.

SEAN currently tracks the following navigation metrics: whether or not the robot reached the goal position, how long it took to reach the final position, collisions with static objects, and the robot's final distance to the goal position. The latter metric is particularly useful for comparison in challenging tasks. In addition, SEAN can continuously track metrics related to social interactions. Currently, we track the closest distance between the robot and pedestrians, as well as the number of collisions with pedestrians, which are recorded separately from collisions with all other objects. These metrics are common in the social navigation literature [12, 15, 17, 22, 24] and serve as a starting point for comparisons among navigation approaches. We plan to expand this set of metrics in the future.

Table 1 provides example results by the Trial Runner for the ROS Navigation Stack [7], which was minimally tuned, and a teleoperated robot. Localization for the ROS Nav. Stack was performed via SLAM [5]. Low performance is attributed to not taking into account human actors during mapping and overly conservative navigation behavior in the dynamic environments [24].

Teleoperation was implemented through a ROS node that connected to a gamepad controller. The teleoperated Jackal did not reach 100% of the target goals because people blocked its way and the episodes timed out. Nonetheless, teleoperation was an interesting baseline for automated methods. It can also serve to gather demonstrations or human preferences for navigation trajectories in the future [11, 25].

As shown by the examples, SEAN is a valuable tool to systematically evaluate performance. Additionally, SEAN can accelerate the development of navigation systems by helping identify problems early on. We do not claim, though, that simulation is a replacement for real-world testing; instead, SEAN compliments real-world evaluation of human-robot interactions in the context of navigation.

3 CONCLUSION & FUTURE WORK

We presented SEAN, a collection of open-source tools and social environments for autonomous robot navigation. Our hope is that the flexibility of the platform encourages researchers to make more comparisons between social robot navigation methods, and more easily study human-robot interactions in the future. For instance, videos from SEAN could be used for qualitative human evaluation of navigation approaches [23]. While we currently provide a limited number of scenes, robots, and evaluation metrics, we plan to expand these features in the future. We also plan to test SEAN in terms of sim-to-real transfer of navigation policies and human evaluation of robot behaviors.

4 ACKNOWLEDGEMENTS

REFERENCES

[1] Alexander Amini, Igor Gilitschenski, Jacob Phillips, Julia Moseyko, Rohan Banerjee, Sertac Karaman, and Daniela Rus. 2020. Learning Robust Control Policies for End-to-End Autonomous Driving from Data-Driven Simulation. *IEEE Robotics and Automation Letters* 5, 2 (2020), 1143–1150.

[2] Anoop Aroor, Susan L Epstein, and Raj Korpan. 2017. MengeROS: A Crowd Simulation Tool for Autonomous Robot Navigation. In *AAAI 2017 Fall Symposium on Artificial Intelligence for Human-Robot Interaction*.

[3] Sean Curtis, Andrew Best, and Dinesh Manocha. 2016. Menge: A modular framework for simulating crowd movement. *Collective Dynamics* 1 (2016), 1–40.

[4] Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. CARLA: An Open Urban Driving Simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*. 1–16.

[5] Giorgio Grisetti, Cyrill Stachniss, and Wolfram Burgard. 2007. Improved techniques for grid mapping with rao-blackwellized particle filters. *IEEE transactions on Robotics* 23, 1 (2007), 34–46.

[6] Winter Guerra, Ezra Tal, Varun Murali, Gilhyun Ryou, and Sertac Karaman. 2019. FlightGoggles: Photorealistic Sensor Simulation for Perception-driven Robotics using Photogrammetry and Virtual Reality. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. https://doi.org/10.1109/iros40897.2019.8968116

[5]https://github.com/Field-Robotics-Japan/unit04_unity/

[7] Rodrigo Longhi Guimarães, André Schneider de Oliveira, João Alberto Fabro, Thiago Becker, and Vinícius Amilgar Brenner. 2016. ROS navigation: Concepts and tutorial. In *Robot Operating System (ROS)*. Springer, 121–160.

[8] Ahmed Hussein, Fernando García, and Cristina Olaverri-Monreal. 2018. Ros and unity based framework for intelligent vehicles control and simulation. In *2018 IEEE International Conference on Vehicular Electronics and Safety (ICVES)*. IEEE, 1–6.

[9] Nathan Koenig and John Hsu. 2013. The many faces of simulation: Use cases for a general purpose simulator. In *Proc. of the 2013 IEEE International Conference on Robotics and Automation (ICRA, Vol. 13)*. 10–11.

[10] Anna Konrad. 2019. *Simulation of Mobile Robots with Unity and ROS: A Case-Study and a Comparison with Gazebo*. Master's thesis. Department of Engineering Science, University West.

[11] Markus Kuderer, Henrik Kretzschmar, and Wolfram Burgard. 2013. Teaching mobile robots to cooperatively navigate in populated environments. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 3138–3143.

[12] Christoforos Mavrogiannis, Alena M Hutchinson, John Macdonald, Patrícia Alves-Oliveira, and Ross A Knepper. 2019. Effects of distinct robot navigation strategies on human behavior in a crowded environment. In *2019 14th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*. IEEE, 421–430.

[13] Wei Meng, Yuchao Hu, Jiaxin Lin, Feng Lin, and Rodney Teo. 2015. ROS+ unity: An efficient high-fidelity 3D multi-UAV navigation and control simulator in GPS-denied environments. In *IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 002562–002567.

[14] Illah R Nourbakhsh, Katia Sycara, Mary Koes, Mark Yong, Michael Lewis, and Steve Burion. 2005. Human-robot teaming for search and rescue. *IEEE Pervasive Computing* 4, 1 (2005), 72–79.

[15] Billy Okal and Kai O Arras. 2016. Learning socially normative robot navigation behaviors with bayesian inverse reinforcement learning. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2889–2895.

[16] Hong Seong Park, Jeong Seok Kang, Hyeong Seob Choi, Sang Woo Meng, and Si Wan Kim. 2013. Simulation-based interface testing automation system and method for robot software components. US Patent 8,601,436.

[17] Ashwini Pokle, Roberto Martín-Martín, Patrick Goebel, Vincent Chow, Hans M Ewald, Junwei Yang, Zhenkai Wang, Amir Sadeghian, Dorsa Sadigh, Silvio Savarese, et al. 2019. Deep local trajectory replanning and control for robot navigation. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 5815–5822.

[18] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. 2009. ROS: an open-source Robot Operating System. In *ICRA workshop on open source software*, Vol. 3. Kobe, Japan, 5.

[19] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M Lopez. 2016. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3234–3243.

[20] Shital Shah, Debadeepta Dey, Chris Lovett, and Ashish Kapoor. 2017. AirSim: High-Fidelity Visual and Physical Simulation for Autonomous Vehicles. In *Field and Service Robotics*. arXiv:arXiv:1705.05065

[21] Samuel S. Sohn, Honglu Zhou, Seonghyeon Moon, Sejong Yoon, Vladimir Pavlovic, , and Mubbasir Kapadia. 2020. Laying the Foundations of Deep Long-Term Crowd Flow Prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*.

[22] Lei Tai, Jingwei Zhang, Ming Liu, and Wolfram Burgard. 2018. Socially compliant navigation through raw depth inputs with generative adversarial imitation learning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*.

[23] Jonas Tjomsland, Sinan Kalkan, and Hatice Gunes. 2020. Mind Your Manners! A Dataset and A Continual Learning Approach for Assessing Social Appropriateness of Robot Actions. *arXiv preprint arXiv:2007.12506* (2020).

[24] Pete Trautman, Jeremy Ma, Richard M Murray, and Andreas Krause. 2015. Robot navigation in dense human crowds: Statistical models and experimental studies of human–robot cooperation. *The International Journal of Robotics Research* 34, 3 (2015), 335–356.

[25] Dizan Vasquez, Billy Okal, and Kai O Arras. 2014. 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*. IEEE, 1341–1346.

[26] Fei Xia, Amir R. Zamir, Zhi-Yang He, Alexander Sax, Jitendra Malik, and Silvio Savarese. 2018. Gibson env: real-world perception for embodied agents. In *Computer Vision and Pattern Recognition (CVPR), 2018 IEEE Conference on*. IEEE.