

Researchers design 'socially aware' robots that can anticipate and safely avoid people on the move

May 18 2022



Hugues Thomas and his collaborators at the U of T Institute for Aerospace Studies created a new method for robot navigation based on self-supervised deep learning. Credit: Safa Jinje

A team of researchers led by University of Toronto Professor Tim Barfoot is using a new strategy that allows robots to avoid colliding with people by predicting the future locations of dynamic obstacles in their path.

The project will be presented at the International Conference on Robotics and Automation in Philadelphia at the end of May.

The results from a simulation, which are not yet peer-reviewed, are available on the arXiv preprint service.

"The principle of our work is to have a [robot](#) predict what people are going to do in the immediate future," says Hugues Thomas, a post-doctoral researcher in Barfoot's lab at the U of T Institute for Aerospace Studies in Faculty of Applied Science & Engineering. "This allows the robot to anticipate the movement of people it encounters rather than react once confronted with those obstacles."

To decide where to move, the robot makes use of Spatiotemporal Occupancy Grid Maps (SOGM). These are 3D grid maps maintained in the robot's processor, with each 2D grid cell containing predicted information about the activity in that space at a specific time. The robot chooses its future actions by processing these maps through existing trajectory-planning algorithms.

Another key tool used by the team is light detection and ranging (lidar), a remote sensing technology similar to radar except that it uses light instead of sound. Each ping of the lidar creates a point stored in the robot's memory. Previous work by the team has focused on labeling these points based on their dynamic properties. This helps the robot recognize different types of objects within its surroundings.

The team's SOGM network is currently able to recognize four lidar point

categories: the ground; permanent fixtures, such as walls; things that are moveable but motionless, such as chairs and tables; and dynamic obstacles, such as people. No human labeling of the data is needed.

"With this work, we hope to enable robots to navigate through crowded indoor spaces in a more socially aware manner," says Barfoot. "By predicting where people and other objects will go, we can plan paths that anticipate what dynamic elements will do."

In the paper, the team reports successful results from the algorithm carried out in simulation. The next challenge is to show similar performance in real-world settings, where human actions can be difficult to predict. As part of this effort, the team has tested their design on the first floor of U of T's Myhal Center for Engineering Innovation & Entrepreneurship, where the robot was able to move past busy students.

"When we do experiment in simulation, we have agents that are encoded to a certain behavior and they will go to a certain point by following the best trajectory to get there," says Thomas. "But that's not what people do in real life."

When people move through spaces, they may hurry or stop abruptly to talk to someone else or turn in a completely different direction. To deal with this kind of behavior, the network employs a machine learning technique known as self-supervised learning.

Self-supervised learning contrasts with other [machine-learning techniques](#), such as reinforced learning, where the algorithm learns to perform a task by maximizing a notion of reward in a trial-and-error manner. While this approach works well for some tasks—for example, a computer learning to play a game such as chess or Go—it is not ideal for this type of navigation.

"With [reinforcement learning](#), you create a black box that makes it difficult to understand the connection between the input—what the robot sees—and the output, or the robot does," says Thomas. "It would also require the robot to fail many times before it learns the right calls, and we didn't want our robot to learn by crashing into people."

By contrast, self-supervised learning is simple and comprehensible, meaning that it's easier to see how the robot is making its decisions. This approach is also point-centric rather than object-centric, which means the network has a closer interpretation of the raw sensor data, allowing for multimodal predictions.

"Many traditional methods detect people as individual objects and create trajectories for them. But since our model is point-centric, our algorithm does not quantify people as individual objects, but recognizes areas where people should be. And if you have a larger group of people, the area gets bigger," says Thomas.

"This research offers a promising direction that could have positive implications in areas such as autonomous driving and robot delivery, where an environment is not entirely predictable."

In the future, the team wants to see if they can scale up their network to learn more subtle cues from dynamic elements in a scene.

"This will take a lot more training data," says Barfoot. "But it should be possible because we've set ourselves up to generate the data in more automatic way: where the robot can gather more data itself while navigating, train better predictive models when not in operation and then use these the next time it navigates a space."

More information: Hugues Thomas, Matthieu Gallet de Saint Aurin, Jian Zhang, Timothy D. Barfoot, Learning Spatiotemporal Occupancy

Grid Maps for Lifelong Navigation in Dynamic Scenes.
arXiv:2108.10585v2 [cs.RO], doi.org/10.48550/arXiv.2108.10585

Provided by University of Toronto

Citation: Researchers design 'socially aware' robots that can anticipate and safely avoid people on the move (2022, May 18) retrieved 26 April 2024 from
<https://techxplore.com/news/2022-05-socially-aware-robots-safely-people.html>

This document is subject to copyright. Apart from any fair dealing for the purpose of private study or research, no part may be reproduced without the written permission. The content is provided for information purposes only.