Dynamic Robot Placement in Smart Environments

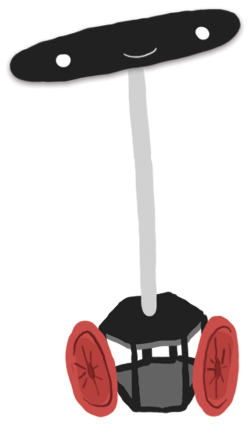
Daylan Kelting, Julia Maliauka, Brittany Manuel, Christopher Pereyda, Aaron Crandall,
Diane Cook, Lawrence Holder and Maureen Schmitter-Edgecombe

Washington State University
Channel Islands California State University

Introduction

- The number of adults over 60 is projected to double by 2050. [1]
- The coming age wave will create an increased need for assistance and a shortage of caregivers.
- Smart home technology and home care robots are a promising way to supplement in-person care.
- To respond to a person's needs more quickly, a robot assistant needs to remain close by.

The goal of this work is to decide where a home care robot should idle when it is not needed.



Methods

Overview

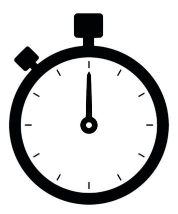
1. Collect motion sensor data from the smart home and generate maps representing human activity over time
2. Input generated maps to an algorithm that selects the robot's resting location
3. Move our robot RAS (Robot Activity Support) to the selected spot

How does the smarthome work?

Motion sensors detect where a resident is in the home.

How often does RAS move to a new spot?

RAS uses the idling algorithm to pick a new spot every hour.
RAS can also pick a "long-term" idle spot overnight, which would correspond with RAS' charging base.



What does the algorithm need to pick a spot?

Our algorithm uses maps generated from the smarthome data to decide where to place our robot, RAS.

The Maps

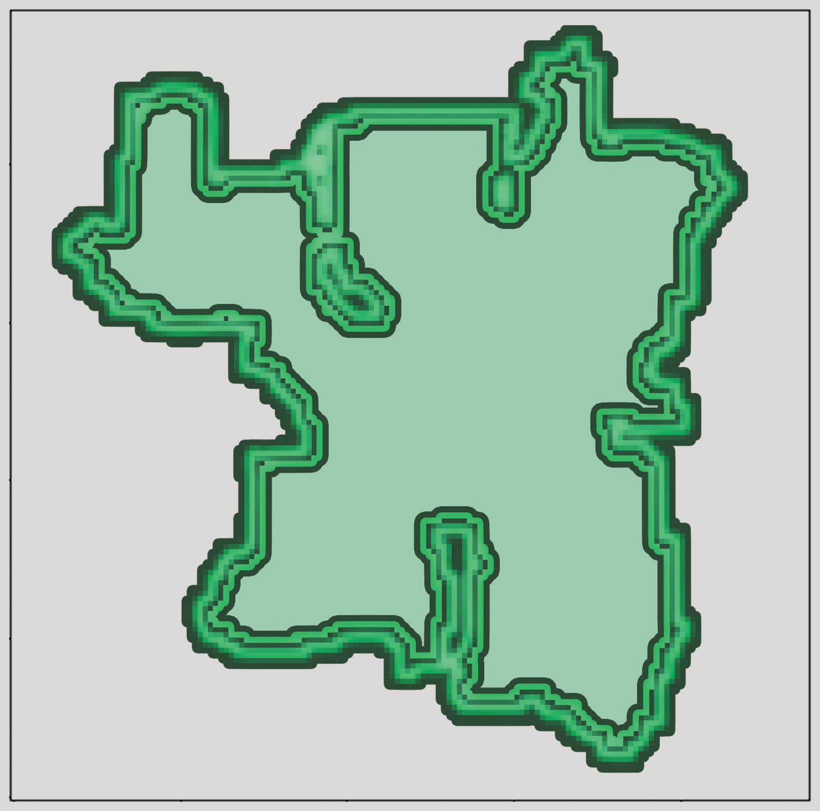
The Heatmaps (long- and short-term)

The Long-Term Heatmap

- Created using motion sensor data from the home.
- Used to determine the resident's traffic patterns.
- The algorithm uses this map to create a pathmap.
- The pathmap visualizes where the resident walks.

The Short-Term Heatmap

- When sensors are triggered, this map updates in realtime.
- The short-term heatmap is meant to represent only the previous hour of resident activity.
- This map is used to generate a weighted average point.
- The algorithm uses this average point to ensure the robot stays close to the resident.



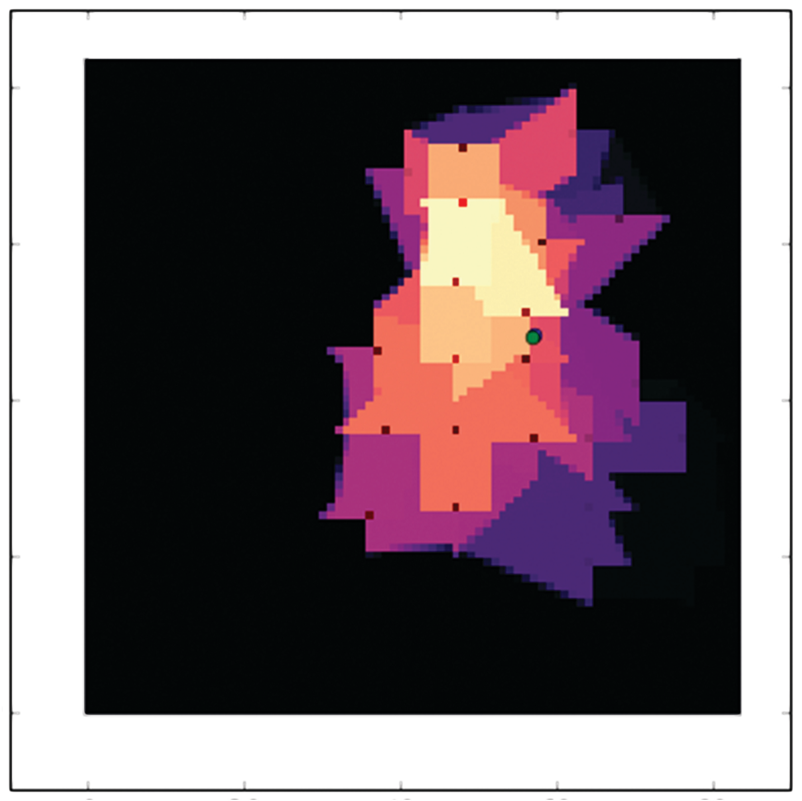
The SLAM map

- The SLAM map identifies walls and obstacles. [2]
- A value in the SLAM map ranges between 0 (empty space) and 100 (wall).
- The SLAM map is used to ensure that RAS does not crash into obstacles.

```
[00 00 00 00 00 00 00 00 00]   [00 00 00 00 00 00 00 00 00]
[00 00 00 00 00 00 00 00 00]   [00 00 00 00 00 00 00 00 00]
[00 00 00 00 00 00 00 00 00]   [12 12 12 12 12 12 12 00 00]
[00 00 00 00 00 00 00 00 00]   [17 17 17 17 17 17 12 00 00]
[00 00 00 00 00 00 00 00 00]   [24 24 24 24 24 17 12 00 00]
[00 00 00 00 00 00 00 00 00]   [34 34 34 34 24 17 12 00 00]
[00 00 00 00 00 00 00 00 00]   [49 49 49 34 24 17 12 00 00]
[00 00 00 00 00 00 00 00 00]   [70 70 49 34 24 17 12 00 00]
[99 00 00 00 00 00 00 00 00]   [99 70 49 34 24 17 12 00 00]
```

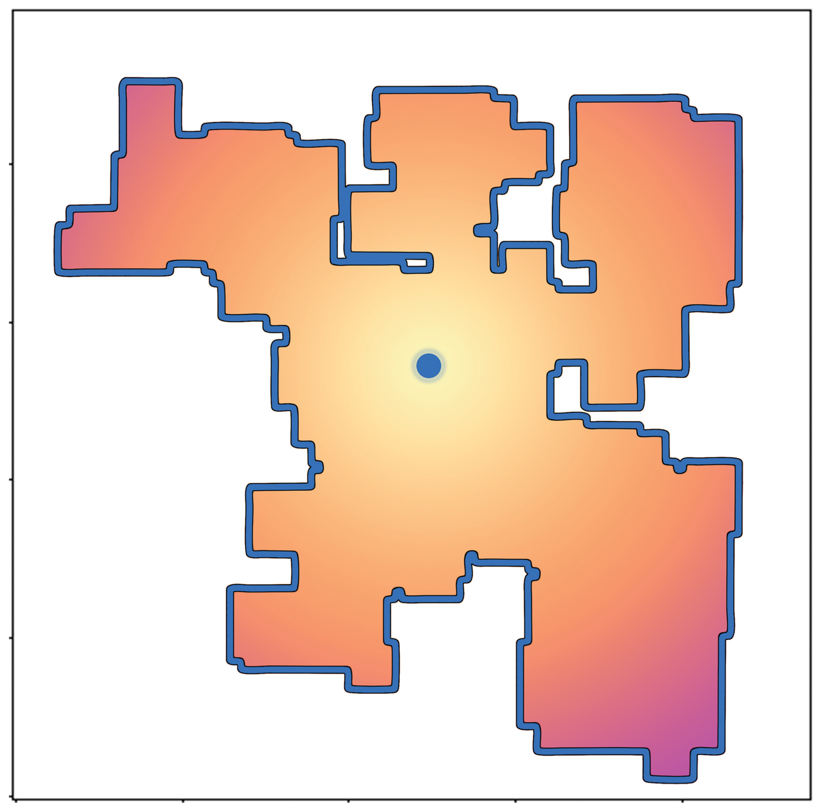
The Wall Map

- The wall map is created using the SLAM map.
- This map encourages RAS to stay near walls.
- In the above matrices, the value of the wall (99) is expanded so that RAS prioritizes residing near the wall.



The Pathmap

- The pathmap is created from the long-term heatmap.
- This map estimates traffic of the resident.
- The algorithm uses this map to try to prevent being "in the path" of the resident to avoid annoying or tripping them.



The algorithm chooses a spot

- The spot stays close to the weighted average location of the resident.
- Avoids being in the resident's way by checking the pathmap.
- Makes sure RAS isn't in a wall or on a couch using the SLAM map.
- In the map above, the yellow regions of the gradient correspond with the algorithm's preferred placement spots.

Results

Historical Results

- We evaluate the algorithm results using historical data collected in the lab over a period of 6 weeks.
- We run this data through a simulator that runs the algorithm and records the average distance to the resident.

Simulator results using our realtime algorithm:
- Average distance to participant: 2.715 meters
- Max distance to participant: 5.375 meters

Results using only historical data placement:
- Average distance to participant: 2.907 meters
- Max distance to participant: 4.000 meters

Participant Study Results

- To compare our algorithm to the best of human performance, we utilize an informal study conducted in the CASAS lab.
- The study asks participants to answer the following questions:
  - how confident they are in their ability to place RAS
  - how familiar they are with the space
- We explain the purpose of our algorithm and ask the participants to evaluate 5 randomly chosen placements.
- We also ask participants to pick up to 2 spots of their own choosing and evaluate them.

Future Work

- We want to generalize our study to include a larger population of participants.
- Participants tend to analyze things that RAS currently has no capacity to see, such as avoiding tall furniture, appliances, and decorations. We want RAS to be able to observe and avoid these objects.
- We would also like to use the depth camera, not just the 2D LiDAR, to help RAS analyze the surrounding area.

Conclusions

Our algorithm is able to generate standby locations that are consistent with placement points suggested by human participants.

Acknowledgements

We'd like to thank Dr. Brian Thomas for his contributions to the RAS project.
This material is based upon work supported by the National Science Foundation REU Program under Grant No. 1757632 and by the National Institutes of Health under Grant No. R25AG046114.

References

[1] Alzheimer's Association, "Alzheimer's disease: Facts and figures," Alzheimer's Dement., vol. 13, no. 4, pp. 325–373, 2017.
[2] W. Hess, D. Kohler, H. Rapp, and D. Andor, "Real-time loop closure in 2D LIDAR SLAM," in IEEE International Conference on Robotics and Automation, 2016, pp. 1271–1278.