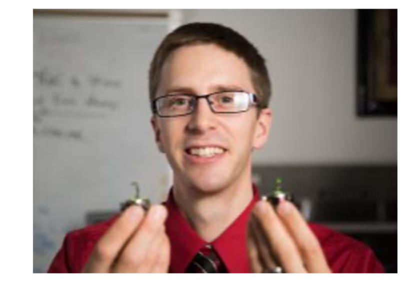
Sensor Implementation in Autonomous Narrative-Capturing Robot

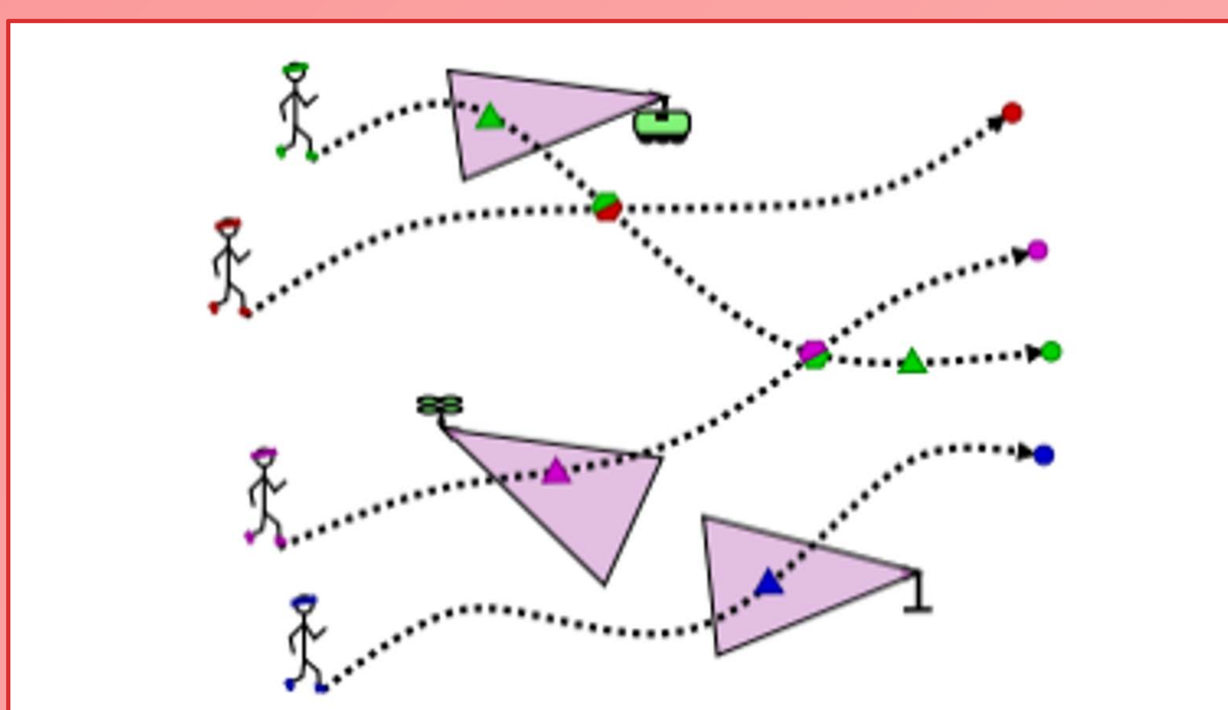




Conlan Taylor, Rhema Ike, Aaron T. Becker
Department of Electrical and Computer Engineering, University of Houston

Motivation and Background

- Narrative storytelling skills, while crucial and commonplace in human interactions, have yet to be developed in advanced robotics applications.
- A novel algorithm will enable an automaton to make decisions when recording a live event – such as a marathon – to capture and string important scenes into a meaningful story.
- The robot's various sensors must collect information about itself, actors of interest, and the environment to guide its decisions.



Robots will track actors and decide which events are worth capturing.
Source: "Planning Coordinated Event Observation for Structured Narratives" [1, Fig. 1].

Objectives

- Write programs that process information from the robot's sensors.
- Test functionality of the sensors.
- Debug and refine the programs.

References and Acknowledgements

[1] D. A. Shell, L. Huang, A. T. Becker, and J. M. O'Kane, "Planning Coordinated Event Observation for Structured Narratives," In Proc. 2019 International Conference on Robotics and Automation (ICRA).

This research was supported by National Science Foundation Grant No. 1849303, and University of Houston's Office of Undergraduate Research Major Awards SURF scholarship.

Special thanks to Mike Yannuzzi from Department of Computer Science, University of Houston, for his mentorship and extensive prior work on the sensor programs.

Special credit is due to our research partners Dylan A. Shell and Diptanil Chaudhuri from Department of Computer Science and Engineering, Texas A&M University, and Jason M. O'Kane and Hazhar Rahmani from Department of Computer Science and Engineering, University of South Carolina, for their work in developing the novel algorithm that makes our project unique.

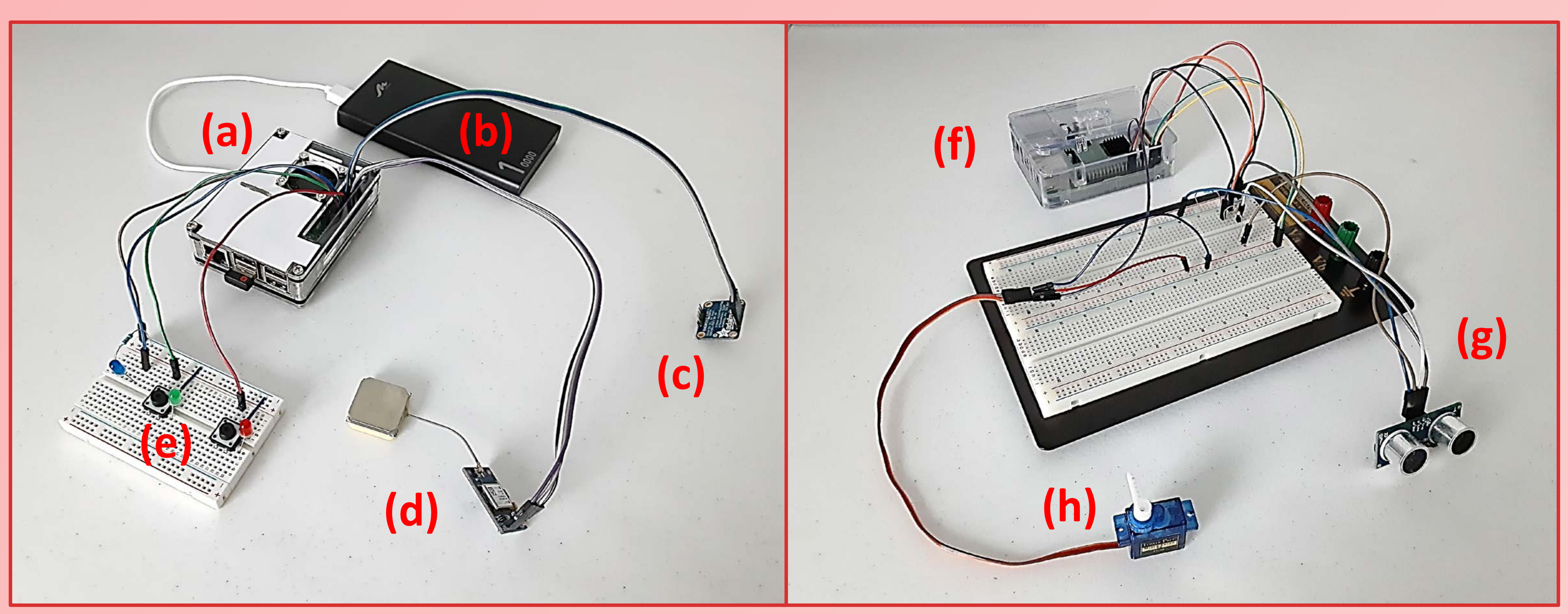
TEXAS A&M UNIVERSITY

South Carolina

Design and Approach

- The robot's body uses an RC car as its base for speed and mobility. The finished model will possess various additional sensors and hardware adjustments.
- Each sensor was tested using Raspberry Pi 3B+. The finished robot will operate using one of the Raspberry Pi 4 family.
- The sensors' code was written in Python.

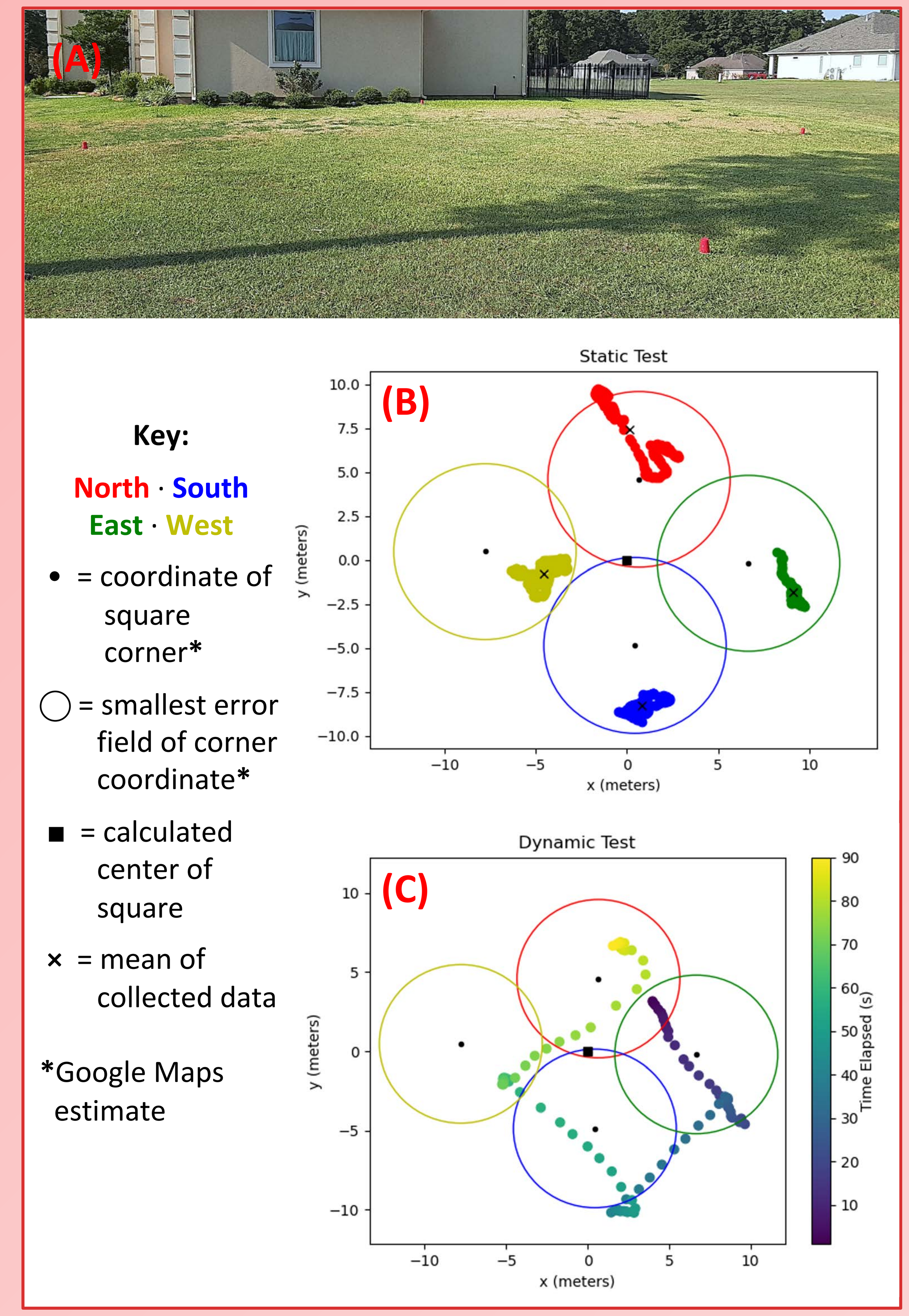
Main body of the robot will be fitted with sensors to interact with its surroundings autonomously.



Setups used to test GPS, IMU, and emergency brake software: (a) Raspberry Pi 3B+ with GPS and IMU code, (b) Miisso 10000 mAh power bank, (c) Adafruit BNO055 IMU, (d) u-blox NEO-6M GPS device with antenna, (e) breadboard fitted with manual push buttons used to activate programs, (f) Raspberry Pi 3B+ with emergency brake prototype code, (g) Adafruit HC-SR04 ultrasonic sonar distance sensor, (h) Tower Pro micro servo motor – standing in for RC robot's motor.

Results

- Completed the prototype for threaded emergency brake program using distance sensor and micro servo motor.
- Debugged and extensively developed the header code modules for IMU and GPS software.
- GPS test results indicated good precision and accuracy, with some predicted minor deviations.



(A) 10m × 10m square plot used to test GPS. Corners marked with red cups; north corner closest to house.
(B) GPS data collected at each corner while stationary for 180 second intervals at each corner.
(C) GPS data collected while walking around perimeter with 10 second pause at each corner; total duration 90 seconds.

Future Tasks

- Test more hardware as it becomes necessary for future experimentation.
- Conduct more detailed field tests with GPS to further pin down accuracy/precision.
- Assist with field tests when robot becomes operational.