Initial Research Report

Project: A Software Application for Making a Robot Autonomous in an OptiTrack Environment

Team Members: Anastasia Reed, Caleb Tietz, Rhea Katari, and Riley Raasch

Description of Project:

Professor William Friedel has built a robot but needs the software to be developed to make the robot autonomous in an OptiTrack environment. The robot is built on a Roomba frame and includes an Arduino and Raspberry Pi to control its movements, as well as a camera on top for visuals.

OptiTrack is a motion capture system that includes specialized cameras, markers, tracking, and software. The cameras can see infrared light and the markers can reflect it. This allows the cameras to track the markers' movement and the software to construct a 3D model of the movement from that provided data. The robot and the OptiTrack environment setup are both located on campus.

The project will require us to build a graphical user interface (GUI) to remotely control the robot and Python scripts to collect the OptiTrack location data, camera data, and automate tasks/control the robot. The GUI will provide an intuitive way to visualize the robot's position in real time and allow for manual control if necessary. Additionally, it will display critical system information such as battery levels, movement commands, and sensor feedback.

The software development process will involve integrating multiple hardware and software components. The Arduino will be responsible for low-level motor control, receiving movement commands from a Raspberry Pi, which will act as the main processing unit. The Raspberry Pi will handle higher-level computations, such as processing camera feeds and communicating with the OptiTrack system. Python scripts will be written to retrieve positional data from OptiTrack, interpret the robot's location, and send movement commands accordingly to navigate autonomously.

Another crucial aspect of the project is testing and debugging the system to ensure seamless integration. We will conduct multiple test runs in the OptiTrack environment to verify the accuracy of movement commands, response times, and tracking precision. Adjustments will be made based on real-time feedback to refine the control algorithms and improve autonomy.

Ultimately, this project will result in a functional, semi-autonomous robot capable of navigating within the OptiTrack system, with a well-structured software stack that allows for both automated and manual operation. This work will contribute to future research in robotics, motion tracking, and control systems.

Background Research on Related Ideas:

While there appears to be no direct research on making a Roomba-based robot autonomous in an OptiTrack environment, similar efforts have been investigated to develop autonomy in other types of robots using OptiTrack as the system can provide very precise location data.

In a paper published by four Electrical and Computer Engineering faculty at New Mexico State University, Qayum et al. (2012), they explored the use of Intelligent Transportation System (ITS) technology to develop and control a remote control (RC) car on a predetermined track using OptiTrack for position and orientation tracking. ITS incorporates various components such as motion sensors, computing power, and communication protocols to enhance traveler experiences by improving safety, efficiency, and collision avoidance. The study aimed to create an autonomous system capable of detecting RC car collisions and reporting them to operators via Zigbee communication protocol, a wireless protocol used communication between clients and servers over personal area networks. The system would then control the car's hardware and employ a virtual vehicle approach, a model simulating the movement and behaviors of a vehicle, for path tracking with OptiTrack.

Similarly, a paper by four researchers, Ahluwalia et al. (2022), at the Institute of Electronics and Computer Science in Riga, Latvia, explored and attempted to address a common challenge in mobile robotics, technology that enables robots to be autonomous: obtaining strong odometry data from sensory information used to estimate changes in position over time. Many deployed mobile robots operate in highly unstructured, partially observable environments and can be difficult for users to interact with. This challenge is further complicated by the structure differences among mobile robots, categorized in the paper as wheeled, tracked, legged, and hybrid. For instance, while wheeled robots were noted to be the easiest to control and localize, tracked robots were prone to slippage, leading to position and orientation errors. To improve odometry accuracy in mobile tracked robots, the researchers developed a robot operating system (ROS)-based mobile robot system using the Jaguar V4 mobile root as a base platform. They tested various visual odometry solutions with different cameras and methods, comparing their results to OptiTrack data. Additionally, an OptiTrack camera system was used for ground truth estimation against three off-the-shelf odometry systems.

Although the robots explored in these papers do not directly influence our project, this research is relevant to our work, as we will be working with an RC-powered robot, command-based navigation, OptiTrack integration, and user experience improvements. In addition, it provides insight into the various types and complexities of mobile robotics.

Initial Ideas and Features for a First Demonstration:

Manually control the altered Roomba robot with a game controller

- Modify a Roomba to accept external control inputs instead of its default cleaning routines.
- Interface a PlayStation game controller with the control system.
- Use a Raspberry Pi or Arduino as the bridge between the game controller and the Roomba's motors.
- Implement motor control logic, including speed adjustment, directional movement, and potential emergency stop features.
- Test the system in real-world conditions, ensuring that the Roomba responds accurately to controller inputs.

Web interface design mockup

- Create a mock-up for user-friendly graphical interface to monitor to potentially control the Roomba remotely in the future.
- Display key metrics such as battery life, movement status, and sensor data (e.g., obstacle detection, camera feed).
- Provide control buttons for manual operation, such as forward, backward, left, and right movement, as well as special commands.
- Decide on what front-end web technologies to use like HTML, CSS, and JavaScript, with frameworks such as React.
- Learn how to use Flask for back-end.

OptiTrack markers installed on the robot

- Attach OptiTrack motion capture markers to the Roomba for real-time tracking.
- Ensure marker placement is optimized for visibility from multiple OptiTrack cameras.

System diagram of how parts work together (Raspberry Pi, Arduino, camera, battery, etc.)

- Design a block diagram showing all major hardware components and their connections.
- Include the Raspberry Pi as the main processing unit, handling communication and control logic.
- Show how the Arduino interacts with the Roomba’s motors and sensors, if applicable.
- Represent the camera module, detailing its function (e.g., streaming video, object detection).
- Indicate the power sources, including battery connections and power distribution.
- Add any additional sensors (e.g., LiDAR, ultrasonic, IMU) if used in navigation.
- Include communication pathways, such as USB, I2C, SPI, or Wi-Fi connections, between components.

Threats and Roadblocks Challenging the Application:

This project has a lot of different components which must all work together to produce a working prototype. One of the most significant challenges will be being able to control the robot remotely in real time. Since this is a custom-built robot, a lot of manual testing will have to be done. There is no documentation or help sheet provided. Additionally, since this is software interfacing with physical hardware, there is always a risk of hardware malfunctions or inaccuracies. Professor Friedel has already mentioned sometimes the two main driving motors will operate at different speeds even though they are programmed to operate at the same rate. Due to issues like this, the team will have to produce work-around solutions such as measuring the number of revolutions a wheel has completed and making live speed adjustments to ensure a straight-line path can be taken correctly.

In addition to the hardware challenges, this application must interact with a server hosted by the robot over the network. Because of this, there may be connection or latency issues which could harm or inhibit the performance of the robot. The team must also design a sort of protocol for how to control the robot. This way, the front-end and back-end can communicate well.

Lastly, one of the main features of the project the team would like to implement is OptiTrack functionality. This could possibly pose a challenge as this is a new system for the team members and will have to be learned. We will have to rely on others to get progress done and are unfamiliar with how much time technologies like OptiTrack will take to develop and interact with. Additionally, we have only recently received our initial starter code from the project, posing a further challenge to our progress so far. Since the OptiTrack lab on campus is a physical location, this will require the team members to be on-site to work on it. The team may want a way to display on the GUI some of the feedback from the OptiTrack lab, so transmitting that feedback to the server hosted on the robot’s Raspberry Pi is another obstacle which may be a challenge to overcome.

Works Cited

Qayum, M. A., Siddique, N. A., Haque, M. A., Tayeen, A. S. Md., & Klipsch School of Electrical & Computer Engineering, New Mexico State University. (2012). Control of autonomous cars for intelligent transportation system. In *IEEE/OSA/IAPR International Conference on Infonnatics, Electronics & Vision*. https://www.sjsu.edu/people/burford.furman/docs/me190/Control%20of%20autonomous%20cars%20for%20intelligent%20transportation%20system.pdf

Ahluwalia, V., Arents, J., Oraby, A., & Greitans, M. (2022). Construction and benchmark of an autonomous tracked mobile robot system. *Robotic Systems and Applications*, *2*(1), 15–28. https://doi.org/10.21595/rsa.2022.22336