Teleoperated Farming Rover

Group: Donavan Erickson, Sam Waddell, Zhuoming Zhang
Sponsor: Kernel Labs
Industry Mentor: Amit Mital
Faculty Mentor: Sam Burden

W ELECTRICAL & COMPUTER ENGINEERING

kernel LABS

The Problem

For a variety of demographic and political reasons, the farming industry is faced with the problem of a diminishing workforce. Automation is one answer to this, but it can be expensive and require heavy maintenance. **How can we create a low-cost, easy-to-use, and easy-to-maintain rover for farmers?**

Our Objective

We designed and built a rover for farm applications, with these goals in mind:

- Creating a robust and general-purpose platform that can handle farm terrain.
- Keeping the product at a very low price -- ~$500 in bulk.
- Allowing the rover to be operated over long distances, so that someone other than the farmer can handle it.
- Creating an easy-to-use platform requiring little set-up by the farmer.

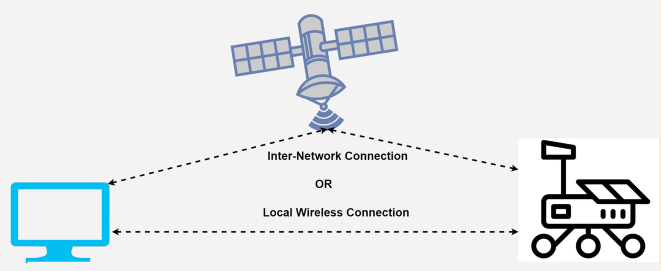

Figure 1: Communication overview

System Overview

Technical Requirements of Our Rover:

- **Mechanical Design**: Capable of driving through tough farm terrain, and towing basic farm tools.
- **Machine Learning**: Framework for future integration.
- **Safety**: If the Raspberry Pi dies, or disconnects from the controller, the rover stops.
- **Teleoperation**: Controlled over LAN or internet for long-distance operation.

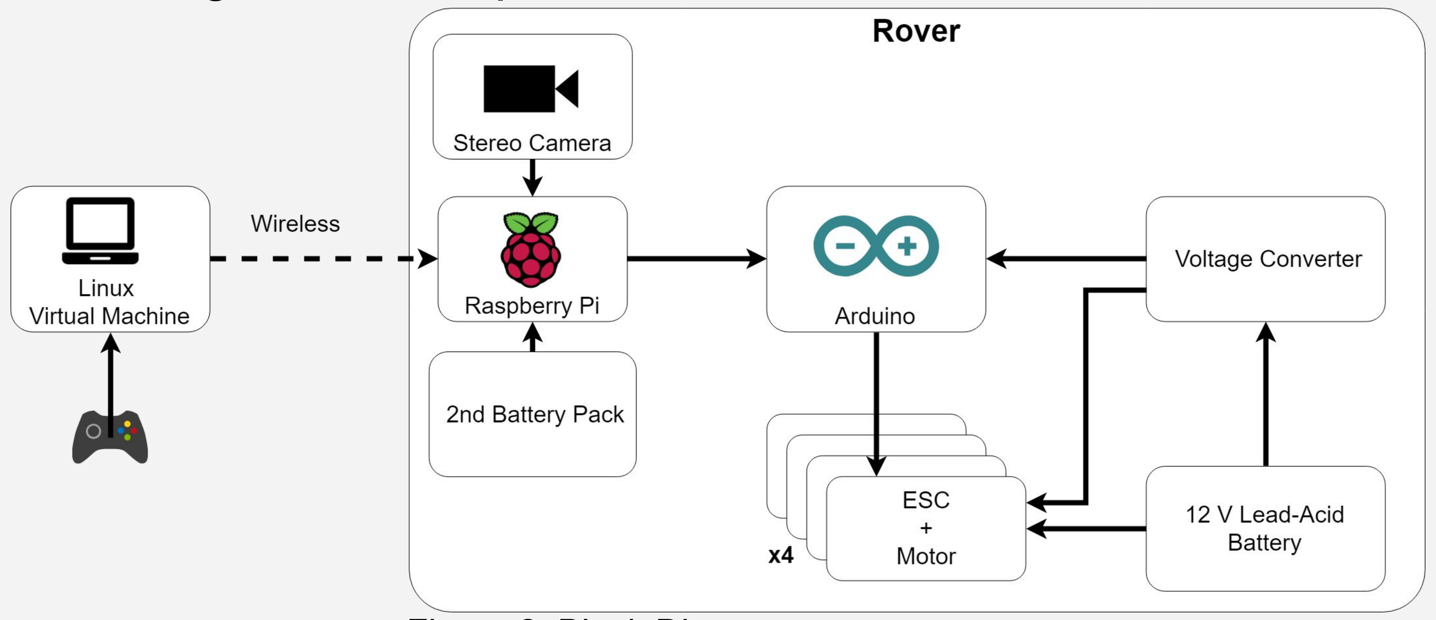

Figure 2: Block Diagram

Implementation

Mechanical Design:

The rover design has gone through many iterations. The original design was a small, direct drive rover and it eventually became a large rocker chassis design. The final design specs are as follows:

- Rocker suspension on front end to keep tires in contact with the ground.
- Four circular saw motors for high speed and torque.
- Go-Kart mud tires to help with farm terrains.
- #35 chain and sprocket design for increased torque and to keep the motors out of the mud.
- ⅝ inch rod to serve a central spine for the robot.
- The front end is made out of aluminum and the back end is made of stainless steel to test the effect of different metals.

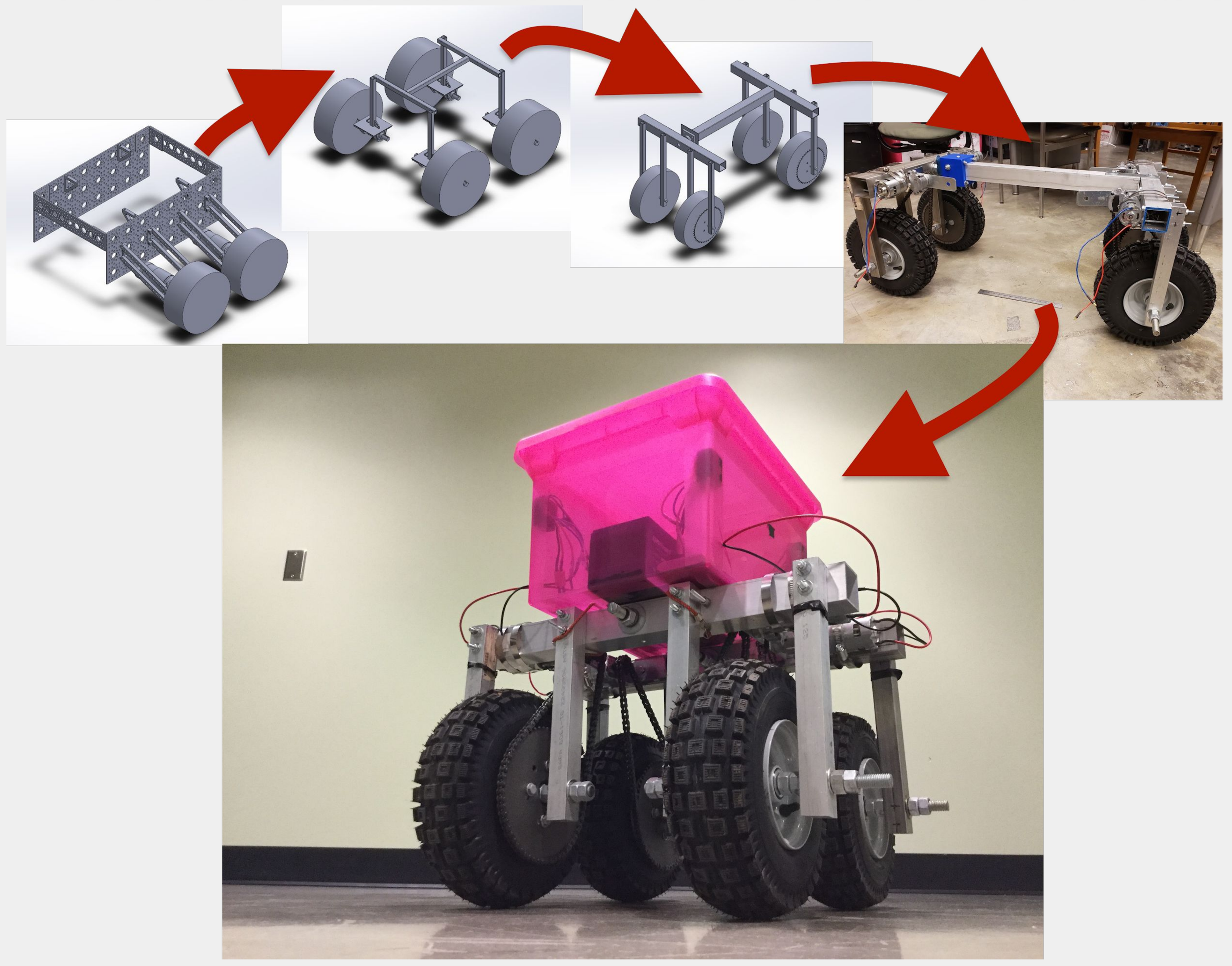
Figure 3: Mechanical Design Progress

Controls:

The controls for the rover use the Robot Operating System (ROS):

- The simple_drive ROS package processes Xbox controller commands that are sent to the Raspberry Pi.
- The Arduino decodes control messages and sends pulse width modulation (PWM) signals to the motor controllers.
- Stereo video is streamed to the operator using the video_stream_opencv and image_view ROS packages.
- Rosbag is used to save stereo image and user input data.

Results & Future Work

Results:

- Current unit of 1 pricing is $974.16 with tax.
- Bulk pricing for 200 robots at a time with suggested BOM is $640 with tax.
- The rover can be controlled by direct connection, or shared LAN with Raspberry Pi, with or without Javascript webpage.
- Virtual machine can collect controller and camera data.


Figure 4: Stereo Camera Image

Future Work:

- Implement Machine Learning for obstacle avoidance and semi-autonomous driving.
- Convert to welded chassis design to reduce parts and facilitate bulk production.
- Add hardware or port-forwarding to facilitate rover's connection to the internet, and enable long range data transfer and control (e.g., using LoRa).
- Improve user-interface for controls.
- Battery and power improvements.

Conclusion

- Robots can help farms dealing with a diminishing workforce, but are often very expensive.
- To combat this problem we created a rover that is low cost, can be teleoperated from long distances, and has the framework to employ machine learning.
- During our time we built a robust prototype that will lay the groundwork for future projects to build on.

Acknowledgement

- Shaiwal Singh, for overall guidance on the project.
- ARUW, for supplying tools and space to operate.
- Matthew Trahms, for help and guidance on machine learning.
- Signey Johnson, for help and guidance on the mechanical design.
- Front desk of Kernel Labs, for always having those mints.