RAILRIDER PROJECT PROPOSAL

By

James Recera

Varun Sharma

Zhanshuo Zhang

Proposal for ECE 445, Senior Design, Spring 2026

TA: Abdullah Alawad

13 February 2026

Project No. 75

Contents

1. Introduction

1.1 Problem

A lot of important industrial inspection locations are basically “thin-structure environments” where a normal robot is awkward or unsafe: narrow beams, cable trays, pipe-rack edges, and long tunnel-like spaces. These places show up in real industrial environments like data centers cable management, HVAC/ventilation runs in industrial facilities, and even some space missions like lunar or martian tunnel scout where falling off an edge could mean mission failure. A typical RC car is too wide and needs turning radius, and a drone can be loud, short battery, and often not allowed indoors nor functioning well in a confined area. We want a compact platform that can stably move on narrow structures and produce some useful inspection results.

1.2 Solution

We will build a reaction-wheel stabilized uni-wheel robot that can travel along a narrow beam/rail while carrying a camera-based perception payload. The core idea is that the robot can balance itself with a tiny contact footprint, so it can ride on structures that would make a 4-wheel robot fall off. Overall, the robot will consist of a drive wheel and a flywheel that rotate orthogonal relative to each other, utilizing conservation of angular momentum to balance itself against gravity. Each of the wheels will be driven by brushed or brushless motors along with motors drivers, and some sensors and encoders monitor and record the rotation speed of the wheels constantly. In order for the robot to stabilize itself robustly, we need to provide enough power with a central battery management system that will deliver power to other subsystems safely, and then implement several robust feedback control loops algorithms on the microcontrollers that receive input information from sensors and output intended voltage to motor drivers.

After achieving the goal for the robot to balance itself on tiny contact, we will implement several computer vision algorithms on the camera-based perception subsystem that communicate with the center microcontroller and output videos to phones or laptops. From these videos, users of this robot could make decisions and remotely control the forward or backward motion of the robot. The camera system with the CV algorithms will have two main functions. Firstly, while the user will drive the robot forward or backward on a narrow path using the software interface we develop on the computer, the camera will detect edges or obstacles on the path and provide a safe override to prevent falloff, so the robot can be driven on any narrow path without complicated human maneuvering. Another function of the camera system is to perform assisted inspections by utilizing some object identification and classification from the computer vision. The robot follows path directions and uses perception cues and logs simple “inspection events” for different types of missions. For example, these events can be broken cables for data centers, debris or blockages for ventilations in industrial facilities, etc.

1.3 Visual Aid

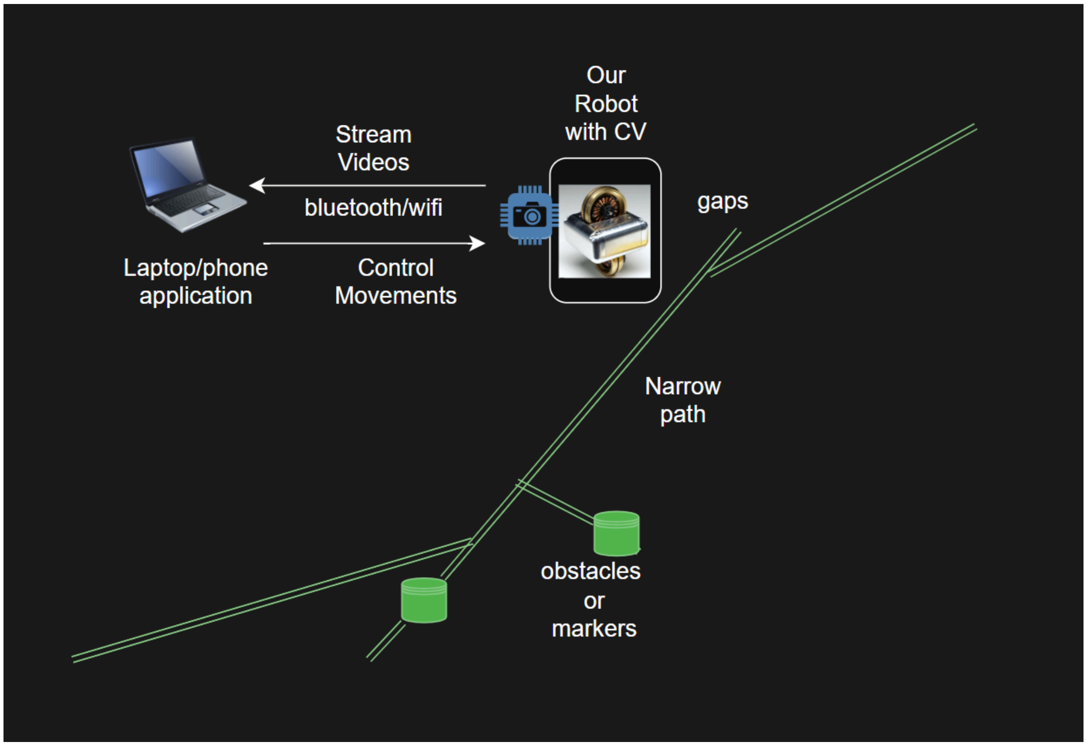

Figure 1. Visual Aids for our Wheelbot project and demonstration

1.4 High-Level Requirements

We pick the three out of four following quantitative characteristics to show success within our project (one of the objective- 3rd or 4th- might be optional depend on the difficulty of the task):

1. Balance: The robot shall demonstrate stable and autonomous self-balancing on a stationary surface for at least 60 seconds without any external support or manual intervention, maintaining an upright attitude throughout the trial.
2. Straight narrow-structure driving: The robot shall be able to drive on a 1 m long straight rail/beam path. The robot, under the user's control, and under self-balancing and stabilizing, should maintain continuous contact with the path without falling off.
3. Twisted narrow-structure driving with branches, gaps, and obstacles that represent the intended operating environment. Under this task the system shall implement a perception-driven safety override that detects hazards like obstacles and gaps along the

path and makes an emergency stop so that the robot comes to rest before the hazard, with a maximum stopping distance of 20 cm.

4. Inspection output: During each drive, the robot shall stream live video to a laptop or phone and generate a structured log containing at least three vision-detected inspection events, such as “marker reached or tag detected,” “edge or drop-off detected and stop triggered,” and “obstacle detected and stop triggered.” If time permits, an additional event type may be included, such as “debris or blockage flagged” or “loose or broken cable flagged.”

2. Design

2.1 Block Diagram

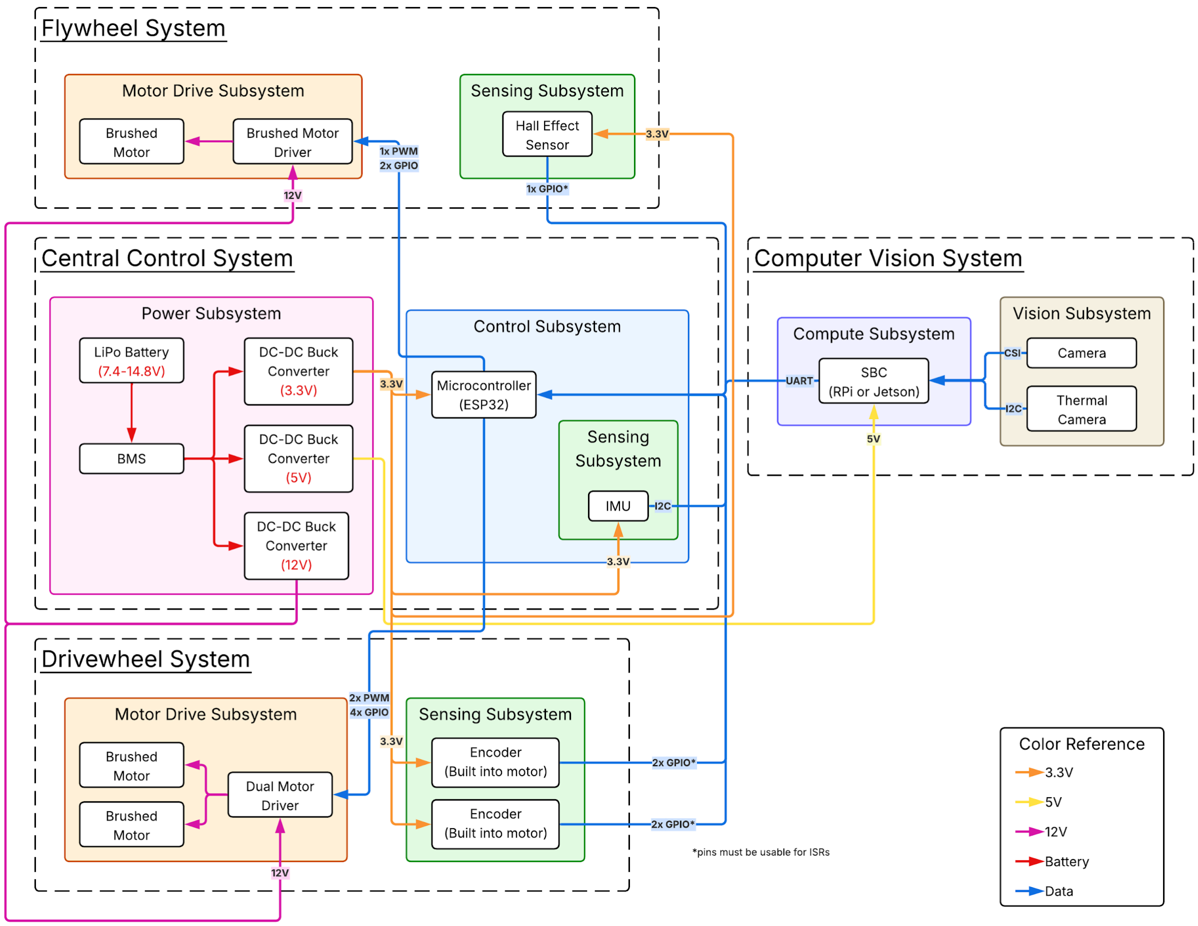

Figure 2. High level block diagram of each system within RailRider

2.2 Subsystem Overview

2.2.1 Motor Drive Subsystem

This subsystem, both contained within the primary drive system and the flywheel system, is responsible for rotating the wheels that allow for the robot to navigate around and to keep itself from tipping over. Each motor is driven by a motor driver that changes the rotation speed of motors from reading the data communicated from MCU, whose calculation was based on data sent from the sensing subsystem (encoders and IMU) that tells each motor how fast to rotate and in which direction to spin.

Requirements

The motor drive subsystem must be fed a consistent 12V, or else the feedback loop being used to control the motors is more likely to fail. Differing voltages, such as that of a battery slowly losing power as energy is consumed, will result in different motor speeds during operation and lead to inconsistent motor performance. Furthermore, higher voltage than the 12V the motors are rated can lead to overheating and early motor failure due to operating out of spec. The motors also must not heat up beyond 60ºC, both for efficiency, safety, and to ensure a 3D printed chassis will not deform while the motors are under load. Each isolated motor drive subsystem must be capable of delivering enough torque to the robot such that when offset up to a maximum of 45 degrees, the robot is capable of pivoting itself back upright about its center of gravity. Motors must also be light enough such that they do not add so much additional weight to the robot that prevents the motors from being able to provide enough torque to pivot the robot upright.

2.2.2 Sensing Subsystem

The sensing system, also contained within both the drive wheel and flywheel systems, must be capable of providing information regarding the motion of the motors to create a feedback loop to drive the motors properly. This system primarily consists of encoders, which allow for the measurement of position and speed of each wheel, as well as an IMU which provides more information about the current rotation and position of the robot. All of this data gets sent back to the control subsystem where it is processed and used within code.

Requirements

For sensing to be effective, data reported back from the IMU must be accurate enough to provide degree precision up to 0.01º, and encoders must have a high enough resolution of at least 4096 counts per full revolution of the robot's wheel. Low resolution and inaccurate data will result in worse precision when running calculations for our feedback loop. Sensors must also be reliable and not fail during operation, and for ease of use with the MCU within our control system, sensors must be capable of running at about 3.3V±0.2V as the GPIO pins for the control system are only able to safely handle up to 3.3V and not 5V.

2.2.3 Control Subsystem

The control system is responsible for intaking data provided from the sensing subsystem, and provides the next time-step control instructions back to the motor drive systems. We do this through the utilization of an MCU, and use control loops in software in order to determine how much power needs to be provided to the motors to bring the robot back to steady state or to drive around in a specific specified direction.

Requirements

The control system needs to run fast enough in order to update the control loop in real time, as well as efficiently take in all incoming data from the encoders and IMU to quickly determine how the motors should rotate to balance the robot. For the balancing feedback loop to be effective, the control system shall run a deterministic real-time loop that is larger than 200 Hz (target 250 Hz) with timing jitter limited to 1 ms per cycle. For the controller to regulate posture accurately, the attitude estimate used in feedback shall provide a tilt resolution of 0.05 degrees and maintain absolute tilt error within ±1.0 degree under stationary conditions, with drift less than 2 degrees over 60 s. Also, the robot should have fast transient response and be able to return up to 2 degrees within 2 seconds to reject disturbances.

We want our control system to be robust enough and remain stable under actuator saturation because the torque provided by our brushed motor is limited. This means we might need to include the anti-windup method if integration action of PID control is used, such that a saturation event lasting up to 0.5 s does not increase the settling time by more than 50% compared to the non-saturated case.

2.2.4 Power Subsystem

Each individual subsystem within our robot requires different amounts of power, which necessitates the need for a specialized subsystem that can provide each of our required voltages. The robot must be fully mobile in order to accomplish meaningful tasks such as traversing down a beam and exploring different areas, thus introducing the need for a battery and a safe way of managing it. Without the power subsystem, components would not be able to receive any power and thus is a key component to the functioning of the rest of the robot.

Requirements

The power subsystem cannot let the battery overcharge or discharge to unsafe levels that would turn it into a hazard. It must also be capable of providing 12V, 5V, and 3.3V as output for the various components throughout the board that operate at these different voltages. It must do so reliably and smoothly without brownouts and excessive fluctuations in voltage. Voltage can vary up to ±0.2V for each respective output. The 12V source must be capable of about 36A of continuous output as it drives the motors, the 5V source must be capable of providing 5A continuously in order to effectively power the SBC under load, and the 3.3V line must be capable of providing 2A continuously to power the MCU and various sensors. All components on the power subsystem must remain reasonably cool and stable throughout operation, and ideally, should have an easy to use interface in order to connect other components to their respective supply voltages.

2.2.5 Compute Subsystem

The compute subsystem is responsible for taking the raw image data provided from the vision subsystem, and processing it in order to determine hazards around the area, and providing meaningful information to the user about what the robot sees around it. It is also responsible for communicating back to the MCU what direction it should drive itself as well as provides a wireless interface for the user to communicate with so they can control the robot.

Requirements

The compute subsystem must be capable of connecting to two different cameras (one thermal and one standard), as well as be fast enough to run real-time computer vision algorithms to identify safe paths and hazards within the area. It must have some form of wireless connectivity in order for users to be able to connect, and contain enough memory and compute capability to both handle a local server for user control and video streaming, as well as accomplishing identified computer vision goals. There must also be a low-latency interface for the compute subsystem to communicate with the control subsystem to send messages back and forth between the two, and said interface must be running at the same voltage to avoid damaging GPIO pins (3.3V).

2.2.6 Vision Subsystem

The vision subsystem is responsible for providing raw image data for the compute subsystem to process. It takes a live image feed of the environment around it and acts as our input sensors for our entire computer vision system.

Requirements

The vision system must be capable of providing both thermal and normal image feeds back to the compute subsystem. It must also be capable of doing so at a minimal resolution of 1920x1080 for the standard image, and 192x192 for the thermal image. Cameras should be capable of connecting back to the compute subsystem with minimal latency so as to provide real time data for the SBC to process.

2.3 Tolerance Analysis

One of the bigger concerns with our design is providing enough overall mechanical force in order to have the robot be able to balance itself, especially under conditions where the robot is off-center. This introduces a general torque requirement for our motor drive subsystem depending on the specific constraints we set for our robot. For our current design, we have set the following physical requirements:

- Must weigh at most 2.0kg
- Must be at most 0.2m tall
- Center of mass must be at the center of the robot, or lower
- Max tilt angle of 45º

Given these constraints, it is possible to derive the total amount of torque required in order to keep the robot upright from one of the motor drive subsystems.

$$\tau = r \cdot F = (\frac{0.2}{2})(2.0 * 9.81 * sin(45)) = 1.669\, Nm$$

Given that this is assuming best case performance of our motors and assuming no mechanical losses, we need to add a safety factor to absolutely ensure that the robot will be capable of pivoting itself. We select a safety factor of 2 to guarantee robot functionality, without overloading the motors. This leaves us with a final required torque value of $3.338Nm$ for pivoting about a single axis and provides a baseline for the selection of components within our motor subsystem.

Looking around, there are motors that when paired together, are capable of providing the torque we require, as well as driver boards that are capable of driving these without getting overloaded or being cost prohibitive. An example motor is the FIT0186 gearmotor [2], capable of delivering $1.8Nm$ of torque at stall individually. Two of these could be paired together to obtain a total of $3.4Nm$ of total torque at stall, well above our required torque with our safety factor included. These draw a maximum current of 7A each and an example motor controller that could drive these is the DFR0601 dual channel motor driver [3], capable of providing up to 12A of continuous current on each channel. Therefore, while the general torque requirement remains a concern within our robot, it is possible to meet these mechanical requirements as long as specific constraints are imposed on the physical design of the robot itself.

3. Ethics, Safety, and Societal Impact

With this project, we have a responsibility to acknowledge certain ethical responsibilities, ensure the safety of others and address the potential societal impacts this robot may have. Especially in the modern age of consumer electronics, there have been increasing concerns surrounding users' privacy, the net benefit these products have in the long term, and ensuring that our project is safe to use both by its users and those who may be exposed to it as well. We have identified and addressed three main points of concerns and how it has influenced the design of the robot to be both ethical and safe.

3.1 Privacy

Current day smart devices often come with a camera built in, usually to provide a feature to users that enhances their experience with the product. For example, smart doorbells may have a camera built in so that a homeowner is able to view who is at their door without having to be present at the door or even inside their own house. However, these devices with cameras are cause for privacy issues, as these companies may be allowed to take footage recorded from these smart cameras and use them to learn more about the user, their habits, and other personal information that users are unaware of. With the design of Railrider, we want to prioritize privacy so user privacy is protected as discussed in the IEEE Code of Ethics [1], and so that no unnecessary data is collected other than to observe the environment the robot is located in. All computer vision processing is done locally on the robot itself in order to ensure recorded video remains local to the user, and any live footage viewed is done on a closed, secure wireless network that only authorized users will have access to. This maximizes user privacy while allowing for Railrider to make use of advanced features, at the expense of needing to have enough compute be locally available on the robot for it to effectively work.

3.2 Safety

While working on and developing Railrider, we have to be mindful of both the mechanical and electrical dangers associated with this project. Two main hazards exist: the flywheel and the battery being used to power the robot.

As the flywheel will be spun to thousands of RPM in order to act as an energy storage device to help balance the robot, this comes with the risk of the flywheel undergoing complete mechanical failure at high speeds, sending hazardous shrapnel towards nearby individuals. In order to mitigate the risk of people getting hurt while the flywheel is undergoing testing, we aim to operate the flywheel in a safe, enclosed environment until the safety of the wheel has been mathematically guaranteed, and to also ensure the final flywheel will be constructed out of a robust material that is able to withstand being spun at high speeds. We will also make sure the flywheel is manufactured precisely to ensure balance and minimize vibrations that would lead to mechanical failure.

With respect to the LiPo battery we plan to use for our project, we plan to follow safe practices when charging batteries such as keeping batteries charging within a battery bag when charging, and keeping our battery stored in a safe, non-flammable location in case of failure. We include a BMS within our project in order to ensure that all battery cells are balanced and not reaching dangerously high or low voltages, and aim to only begin using such a battery within our project only when we know the general operation of our robot is achievable on a bench power supply. We also aim to add a checklist for the robot whenever we plan to begin using a battery with our system, checking wiring, boards, and contacts around the robot to guarantee it is safe to plug the battery in before doing so. This thereby mitigates the amount of danger introduced to ourselves and those around during the testing and development process.

3.3 Public Well-Being

As technology progresses, there comes the fear of new machines and innovations rendering certain jobs obsolete. For instance, manufacturing throughout history has gone from having an assembly line of human workers to instead now having machines that are capable of assembling a car from beginning to end with little human intervention. With this in mind, there is reason for concern that a project such as Railrider may pave the way for certain human inspection jobs to become obsolete or useless. We wish not for our robot to fully automate the human inspection process of certain environments but rather for it to act as another tool people can use in unideal conditions - tight, narrow spaces, thin beamed structures, areas where human deployment is not feasible, etc. The compute capability we aim to deploy on the robot, while we aim to make it capable and performant, will still not fully replace the experience and knowledge a human has when identifying environmental hazards. The robot will still rely on a human operator to help use Railrider as a tool, and we hope that this enhances the job of inspectors and allows for them to explore more areas that previously were not traversable. Thus, we aim to improve the quality of life for users of the product, rather than hindering it through innovation.

References

[1] IEEE, "IEEE Code of Ethics | IEEE," *Ieee.org*, 2020.
https://www.ieee.org/about/corporate/governance/p7-8
[2] FIT0186 Dfrobot | motors, actuators, solenoids and drivers | digikey,
https://www.digikey.com/en/products/detail/dfrobot/FIT0186/6588528 (accessed Feb. 14, 2026).
[3] DFR0601 dfrobot | motors, actuators, solenoids and drivers | digikey,
https://www.digikey.com/en/products/detail/dfrobot/DFR0601/10279757 (accessed Feb. 14, 2026).