NumaRover: Autonomous Rover for Delivery

Adrian Cuevas   David Nguyen   Eibi Perez   Palamy Thepouthay

August 20, 2023

Abstract

For this project, we propose NumaRover, a combined autonomous vehicle and self-delivering system able to navigate through a desired path and deliver packages to its destination. To accomplish this, the project employs methods from the fields of computer vision, distributed systems, and internet of things. The project consists of a multi-step process: extract features from visual data, send data to all servos, and avoid objects based on visual cues.

Department of Computer Science and Engineering

University of Arkansas – Fort Smith

Contents

1 Introduction

The following sections will provide more information regarding the problem and proposed solution.

1.1 Problem Statement

Mail carriers everyday have a set day and time they must make the deliveries. The problems that they encounter may be as simple as time. They may be running behind on making their deliveries because of high traffic or some obstacles obstructing their path. [1] talks about how mail carriers face stress when there are delays in delivering mail. Imagine if this happens for every delivery they make. There would be a bottleneck on incoming packages waiting to be delivered, while angering people waiting on their packages to be delivered. There is no time to waste in a career where time is a major factor.

Another issue while making deliveries is that there may be too much for one person to carry. This can include an abundant of packages to be delivered to one location. Another problem may be that the package is too heavy for a person to carry. In [2] it also says that making deliveries is a physically demanding job. If some people are not comfortable with carrying heavy things, then they will not be able to make that delivery, or their body will be sore/hurting.

Faculty members at UAFS, like mail carriers, may need to deliver something to other faculty members. They may face similar problems like mail carriers are experiencing like time and package weight limits. Not only is it tiring but it is also time consuming especially if the destination is on the opposite side of the campus. For example, when students need to take a test in the ASC, the professor must hand deliver the test to the student. The professor cannot email the test to the student since the student may share the test to other students within the class. As mentioned before, the professor may be too busy or have their time wasted walking a long distance on campus. They will have to take time out of their schedule to go make this necessary delivery.

1.2 Objective

The goal of this project is to make a self-driving rover that can aid in delivering packages within the computer science department. On its deliveries, it will detect obstacles along the way and make necessary decisions to get to its destination. Some obstacles that may come up are people, chairs, or trash cans on the rover's path. The rover will make decisions to go around an obstacle. While our prototype rover may not hold heavy objects, this project will be a proof of concept that it can go to a set destination and back to where started (i.e., AI lab to the break room).

Achievement of this objective is predicated on the completion of the following sub-goals:

1. **Construct dataset**. A custom dataset consisting of images captured from the Baldor building. The images are manually labeled with the target values consisting of steering value, object detection, and bounding region.

2. **Identify location, steering, and object**. A neural network with object detection capabilities is implemented to determine the location, and steering values of the servos.

3. **Building the rover**. Create and assemble all printed parts along with the metal bars and axles and put together to form NumaRover.

4. **Assemble hardware**. Connect and wire all electrical hardware components, be able to communicate with each other through the Jetson Nano.

2 Background

The following sections will provide the necessary background and information regarding the technology to be utilized.

2.1 Overview

In this project, our goal is to create an autonomous driving rover utilizing visual data from a real-time camera in order to navigate to its delivery location. One of the AI models that we are considering using is a Convolutional Neural Network (CNN). CNN is a deep learning model for processing data that has the ability to classify image data. The main idea of CNN is to use a special layer called a convolution layer that extracts parts of the image, by doing this the model can learn image features efficiently and accurately. In this case, we will use the system to control the steering in the right direction. In addition, to avoid obstacles during path navigation, we implemented the You Only Look Once version 7 model (YOLOv7) [3] model. The YOLO model is a model used to detect objects from video or pictures. It can also be used for real-time object detection based on the data stream. It takes an image as input and then uses a convolutional neural network to detect objects in the image while creating bounding boxes for the objects the model detected. In this case, it will be used a checkpoint detection.

2.2 Technology

The following sections will provide the necessary software applications, languages, and libraries regarding the technology to be utilized.

1. **Jetson Nano**. The NVIDIA Jetson Nano is a development board used to create projects based on neural networks, deep learning, and AI. With this, Jetson can create a wide range of projects from small smart IoT applications to robots. More complex motors, artificial vision, object recognition systems, devices that intelligently respond by evaluating a set of sensor parameters, small autonomous vehicles, etc.
2. **Webcam**. The purpose of the webcam is to collect images of the rovers current path and feed the live images to our models to determine what do next.
3. **Servo**. Servos are responsible for the motion control.
4. **Python**. Python will be our main programming language to control our rover. Python is a programming language widely used in web applications, software development, data science, deep learning and machine learning. Python is used by developers because it is powerful, easy to learn, can work on a wide variety of platforms, and has many libraries available.
5. **Scikit-learn**. An open-source of libraries and algorithms that focus on the field of machine learning, artificial intelligence, and deep learning. Which is involved in data modeling to predict/regress various types

6. **TensorFlow**. It is a library for building open source Machine Learning Models from Google.
7. **Keras**. It is a Python neural network library that supports data processing.
8. **PySerial**. It is a Python package that allows us to exchange data between computers and pieces of hardware such as servos, voltmeters, flow meters, lights, etc.
9. **Jetson GPIO**. Jetson GPIO is a Python library that provides interactions with the GPIO pins.
10. **OpenCV**. OpenCV is a library developed by Intel for the development of open-source systems that can be used to build Machine Learning or AI to detect and recognize faces, identify objects, classify human actions in videos, track camera movements, track moving objects, extract 3D models of objects, and etc.

2.3 Related Work

Autonomous Delivery Rovers are another innovation that has transformed the business sector and meet consumers' digital lifestyles. This technology has gained a lot of attention from innovators. Therefore, there is continuous development and experimentation. In this project, the hardware parts, we have adapted our design from GitHub repository called "Sawppy the Rover"[2], this rover is a recreation of the Mars rover with this design having a price target of around $500. On the software side, there is a large amount of literature on autonomous robots has been published, we found one of the AI models that we aim to use for navigating autonomous delivery bots, such as convolutional neural networks (CNN) [4]. The performance of CNN have been widely used in imaging task, we plan on leveraging CNN models while applying it in object detection and path prediction for our implementation.

3 Design

The following sections will provide information regarding the requirements and analysis of the project.

3.1 Project Requirements

1. 3D printed parts and aluminium extrusions - utilized to build NumaRover
2. Jetson Nano - utilized to communicate with all hardware components
3. Linux Computer - utilized for running the project and handling models
4. Electrical components - connect all hardware parts
5. GPU - accelerate training
6. Gitea - collaborative space

3.2 Architecture

The architecture will utilized a distributed system. The Nivida Jetson Nano will be receiving the input from the webcam and sending the webcam feed to the computer. The computer will then send the webcam feed to the AI models. The self-driving model and object avoidance model will predict the movement of the rover and then output the steering values to the Jetson to control the servos. The Checkpoint Detection (YOLOv7) model will be used as checkpoint detection to keep track of the rover position.

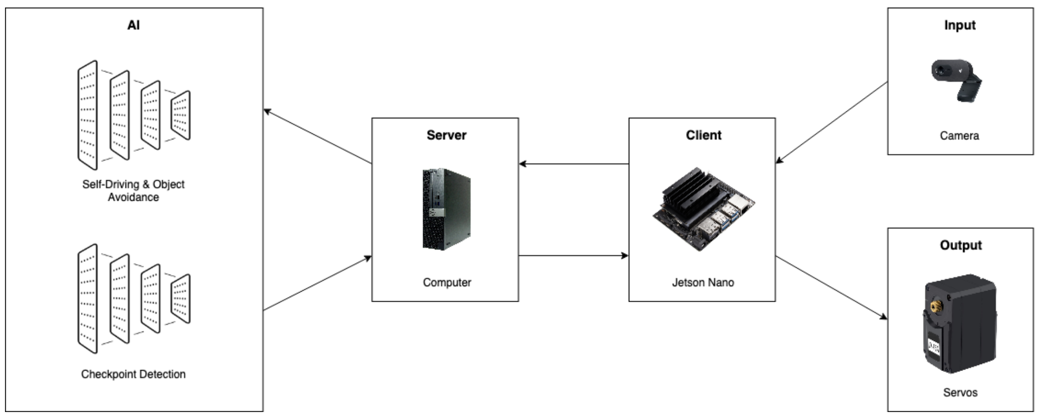


Figure 1: This diagram shows the overall architecture of the rover.

The figure below represents the flow of action when the rover is ready to go. First the rover will be loaded with a package and then the sender will send off the rover to its destination with their hand on the sensor for five seconds to confirm delivery. On the rover journey if it detects an object within the path, it will avoid it, if it does not detect an object than it will check to see if it has reached a checkpoint. From there if the rover detects a checkpoint, it will send out a checkpoint message, if it does not detect a checkpoint than it will see if an object is within its way. Finally if the destination has been reached, the rover will stop and messaged out the destination, if the checkpoint is not the destination than it will loop all the way back to see if an object is within its way. The user can put their hand over the sensor if the recipient wants to send the rover back to its destination.

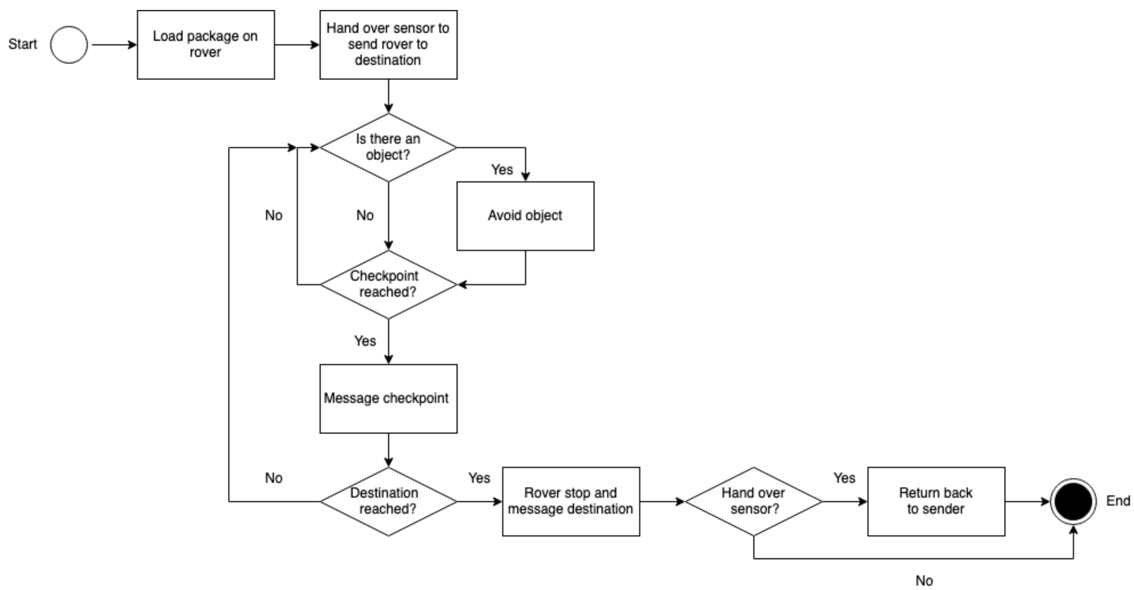


Figure 2: This flowchart shows the flow of the rover actions.

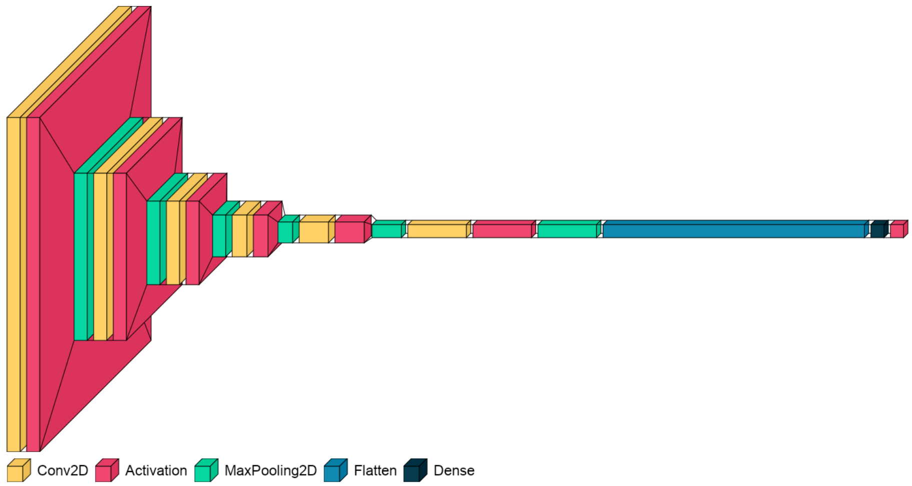


Figure 3: This figure show the custom convolutional neural network model used for self-driving

3.3 Risks

1. **Longevity of the parts** – To prevent possible severe wear of the rover, we will inspect the rover after every test run we do. Some examples will be checking the bolts, checking the servos, checking the wiring, etc.
2. **Data collection** – Due to the Federal Educational Rights and Privacy Act (FERPA), we will collect data when there are no students on campus.
3. **Data overfitting** – Data argumentation and distinguishing turns will be used to avoid the rover making a wrong move.
4. **Physical theft** – To prevent malicious people from physically stealing the rover, there will be a checkpoint system where the rover will update it position on campus.

3.4 Tasks

1. Itemize list of parts to build an autonomous rover
2. Create a list of parts to print
3. Create a timeline of how long to assemble the rover
4. Create a timeline to estimate time to print parts of the rover
5. Begin test printing rover parts
6. Begin printing rover parts
7. Explain the basics of CNN models to teammates

8. Begin experimenting with prototype, models, and datasets
   (a) Build prototype
   (b) Build dataset
   (c) Begin trying different models
9. Test models to see which models/approaches work best
10. Begin assembling the final product
11. Configure sensors, Jetson nano, and motors
    (a) Configure rover for basic movements
12. Collect dataset for the final product
13. Label dataset
14. Retrain models on new datasets
15. Test models
16. Fix any bugs
17. Run multiple delivery/test cases
18. Construct final presentation
19. Construct final report and finalize project

4 Experimentation

4.1 Self Driving

A convolutional neural network was constructed to predict the steering value of the servos based on the image it used as input. This network consisted of several convolutional layers followed by max pooling layers, using the linear activation function. The encoded image would then be passed into a dense neural head to predict the steering value. The last dense layer had a linear activation function since the task was a regression-based task. Adjustments in the network's architecture included the frequency of layers and the number of neurons per layer, kernel size, input size, color images, batch normalization.

4.2 Checkpoint Detection

A test dataset was used with colored blocks which served as checkpoints for the JetsonRacer. As the project continued, the colored blocks no longer served its purposed since the rover was much larger than the JetsonRacer. The new checkpoints are pieces of paper with numbers indicating its position within the building. For this model YOLOv7 was used. YOLOv7 takes in consideration a 'confidence level' to determine the presence of an object within a bounded region. We used YOLOv7 to bound the checkpoint paper while labeling the image with the corresponding class.

5 Results

5.1 Self Driving

The self-driving model along with the object avoidance model trained across 200 epochs with 80% of the data reserved for training and 20% for testing.

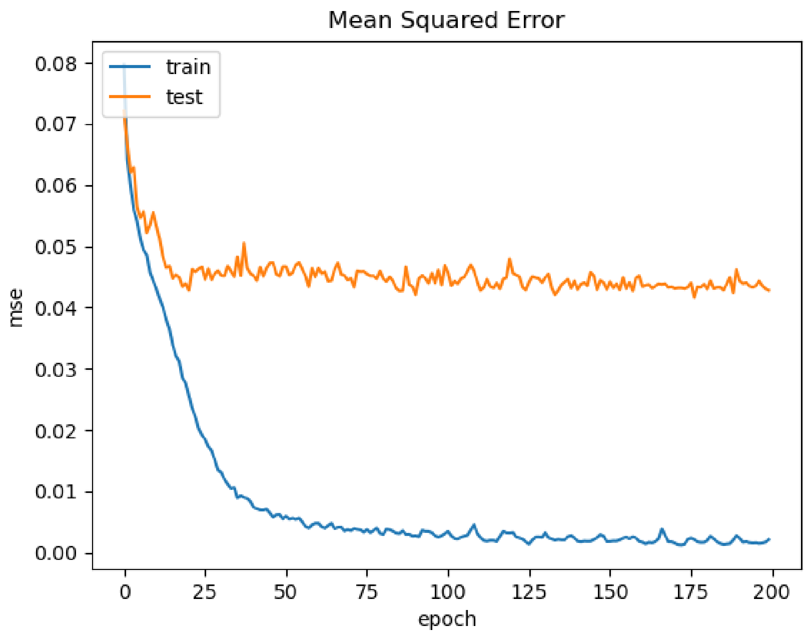


Figure 4: This chart shows the loss over time with the number of epochs.

5.2 Checkpoint Detection

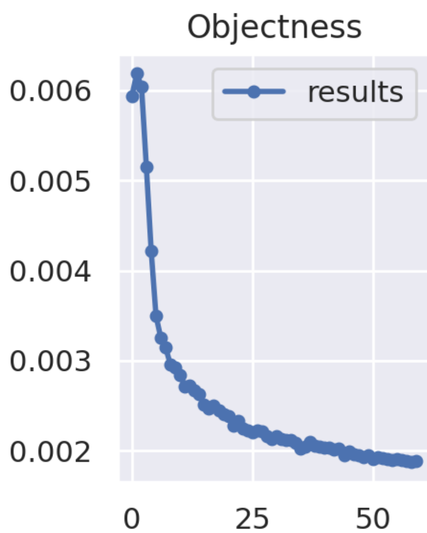


Figure 5: This chart shows the results of the objectness of the model.

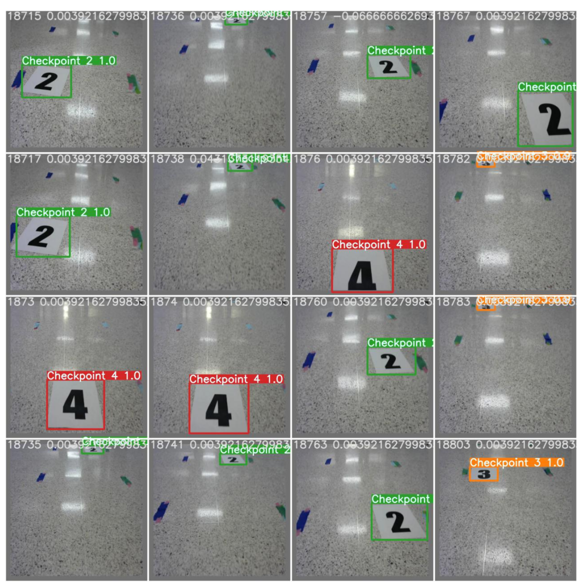
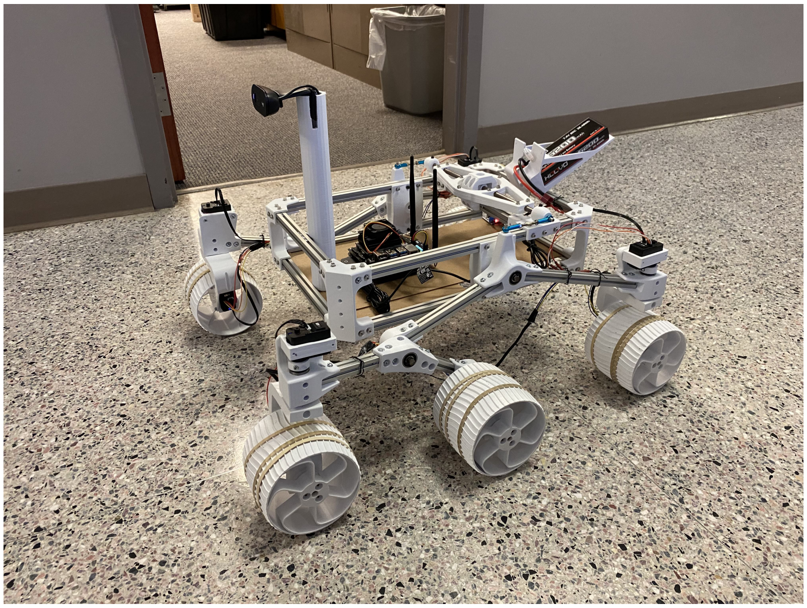
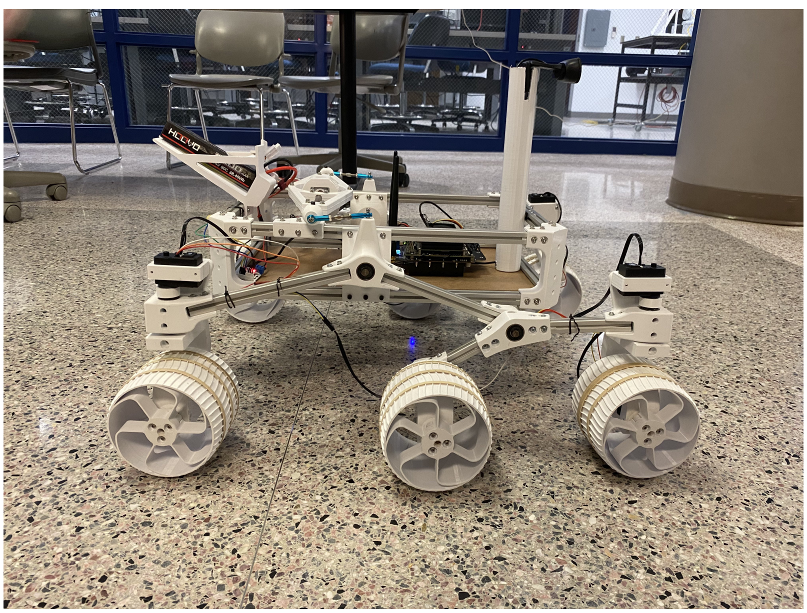


Figure 6: This picture shows the predicted results on test pictures.

5.3 Screenshots





5.4 Schedule

Task	**Assigned**	**Start**	**End**
Project Proposal	Everyone	Week 1	Week 2
Itemize list of parts to build autonomous rover	David,Adrian	Week 1	Week 2
Create list to parts to print	Adrian	Week 2	Week 2
Create a timeline of how long to assemble and print rover	David	Week 2	Week 2
Begin test printing parts	Adrian	Week 3	Week 3
Explain basics of CNN to teammates	Everyone	Week 3	Week 3
Build prototype	Palamy,Eibi	Week 3	Week 3
Begin printing rover parts	Adrian	Week 3	Week 3
Collect test dataset	Palamy,Eibi	Week 4	Week 4
Label test dataset	Everyone	Week 4	Week 4
Experiment on object detection (using YOLOv7) and discord notification	Palamy, David	Week 5	Week 8
Experiment on self driving model	Adrian	Week 5	Week 8
Test self driving model	Eibi	Week 5	Week 8
Experiment on the object avoiding	Eibi, David	Week 5	Week 8
Experiment on servos and connect servos with Jetson Nano	Palamy, Adrain	Week 8	Week 10
Distributed system	David	Week 8	Week 10
Prep printed parts	David,Eibi	Week 10	Week 10
Begin Assembling final prototype	Everyone	Week 10	Week 13
Configure basic rover sensors	Everyone	Week 13	Week 14
Collect dataset for final prototype	Everyone	Week 14	Week 15
Label dataset	Everyone	Week 15	Week 15
Integrate button in the rover	Palamy, Eibi	Week 15	Week 15
Retrain models	David,Adrian	Week 15	Week 15
Test models	David,Adrian,Palamy	Week 15	Week 15
Run test cases	David,Adrian,Palamy	Week 15	Week 15
Fix any bugs	David,Adrian,Palamy	Week 15	Week 15
Construct final presentation	Everyone	Week 14	Week 15
Final Report	Everyone	Week 15	Week 16

5.5 Deliverables

Provide an itemized list of all items to be delivered upon completion of this project.

1. Project Proposal
2. Rover Design Images
3. Itemized list of parts needed to build rover
4. Itemized list of parts needed to print
5. Machine Learning Model Code (in Jupyter Notebook Format)
6. Final rover prototype
7. Final Report

6 Project Members

In this section, all project members and advisors are presented.

6.1 Team Members

Adrian Cuevas (4023) Adrian Cuevas is a Computer Science major with a concentration in Data Science/Artificial Intelligence and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas - Fort Smith. Relevant coursework completed for this project includes CS 3113 – Artificial Intelligence, CS 3323 - Computer Graphics, CS 3103 – Algorithms, CS 4343 – Natural Language Processing, CS 4143 –Deep Learning. He also obtained relevant experience through his work as a Department Tutor and Lab assistant at UAFS. In addition, he obtained research experience as a member of the UAFS Artificial Intelligence Research Lab. He will also be responsible be assisting in data labeling, training, evaluating, and experimenting with the deep learning models we plan to implement.

David Nguyen (4023) David Nguyen is Computer Science major in the department of Computer Science and Engineering at the University of Arkansas - Fort Smith. He has completed relevant coursework for the proposed project by completing CS 3103 - Algorithms, CS 3113 - Artificial Intelligence, and CS 4343 - Natural Language Processing. His responsibilities will include making the parts list, building the rover, creation of the distributed system, and development of the checkpoint model.

Eibi Perez (4023) Eibi Perez is a Computer Science major and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas - Fort Smith. He has coursework that will aid in the proposed project by completing CS 3113 - Artificial Intelligence, CS 3323 - Computer Graphics, CS 4333 - Machine Learning, and CS 4363 - Internet of Things Development. His responsibilities will include the development of a machine learning model.

Palamy Thepouthay (4023) Palamy Thepouthay is a Computer Science major with concentration in Data Science and Artificial Intelligence and a minor in Mathematics in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith. She has completed relevant coursework for the proposed project by completing CS 3113 – Artificial Intelligence, CS 4343 – Natural Language Processing, and CS 4363 – Internet of Things Development. She also obtained relevant experience through her internship at ArcBest Technologies as a Software Development Intern. Her responsibilities will include the development of a machine learning model.

6.2 Departmental Advisors

Our project is supervised by Professor Andrew L. Mackey and Professor Israel Cuevas in the department of Computer Science and Engineering at the University of Arkansas – Fort Smith.

References

[1] Z. Expert, "What are the pros and cons of being a mail carrier?" Nov 2022. [Online]. Available: https://www.zippia.com/answers/what-are-the-pros-and-cons-of-being-a-mail-carrier/

[2] GitHub, " Sawppy Rover," Aug 24, 2022. [Online]. Available: https://github.com/Roger-random/Sawppy_Rover/blob/main/docs/README.md

[3] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, "YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors," *arXiv:2207.02696 [cs]*, Jul. 2022, Available: https://arxiv.org/abs/2207.02696

[4] Jijesh, J. J., et al. Development of Reduced Human Intervention Delivery Robot for Covid-19 Pandemic. No. 6332. EasyChair, 2021.