An Automated Transport System for Industrial Applications

Neel Gondalia | Shawn Hustins | Vaishnavi Kotha | Kye Morris–Takamoto | Cesar Wen Zhu

BotNINJA Technologies

PROBLEM STATEMENT

Modern warehouses are quickly implementing fleets of automated robots to transport packages.

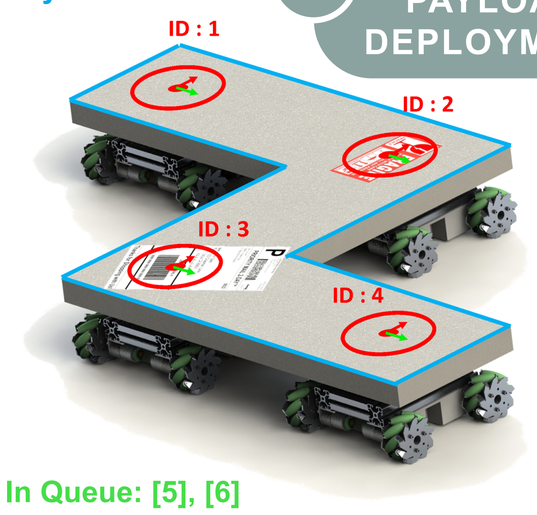
As carrier robots are individualized to haul packages of a specific shape and size, issues arise when a warehouse must handle a range of package sizes.

The goal is to design a dynamic swarm robotics system that can coordinate the transport of packages of varying sizes to a specified location.

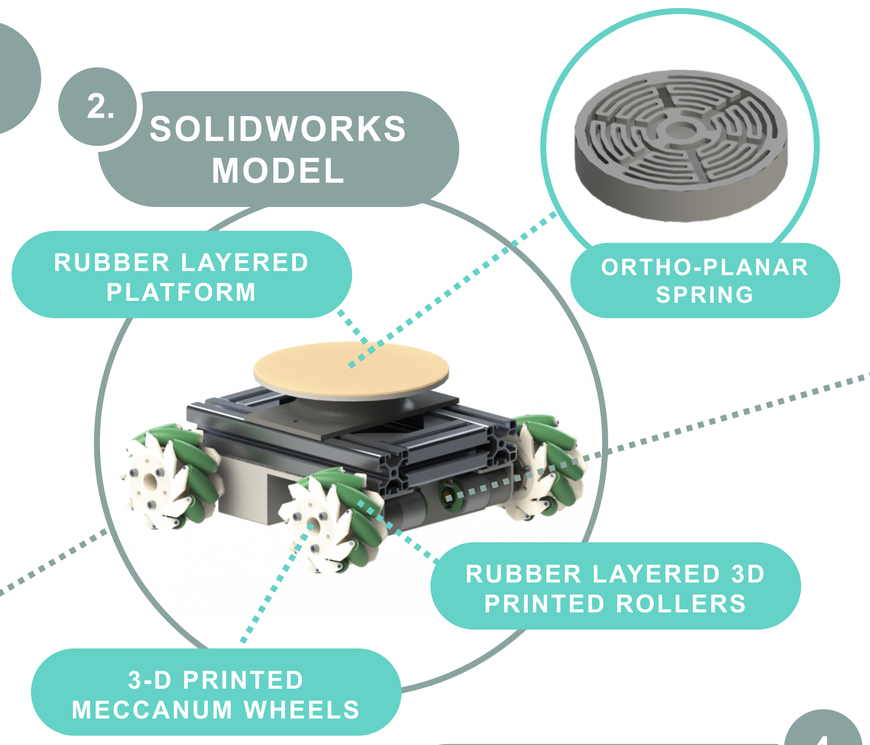
OBJECTIVES

- Determine how many robots will be needed for delivery.
- Coordinate transport of needed number of robots to package retrieval site.
- Recognize payload has been transferred to swarm robot system for hauling.
- Coordinate delivery of given package to specified location within space.

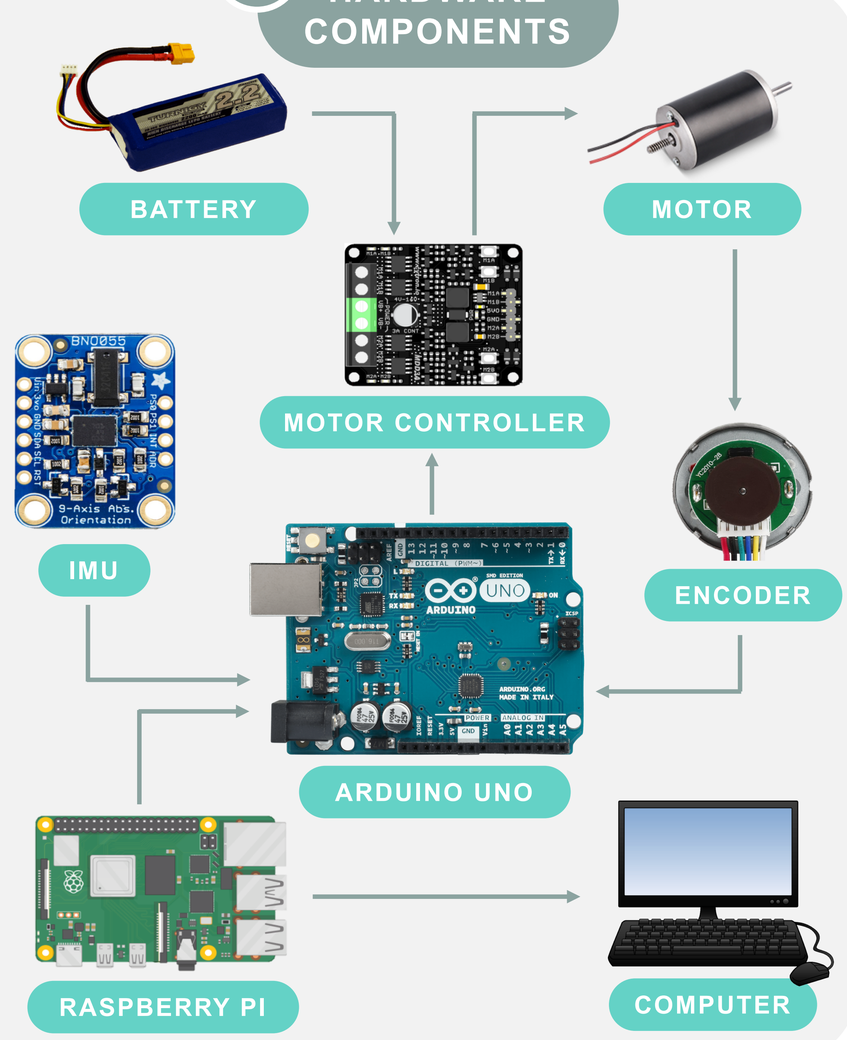
1. PAYLOAD DEPLOYMENT



2. SOLIDWORKS MODEL



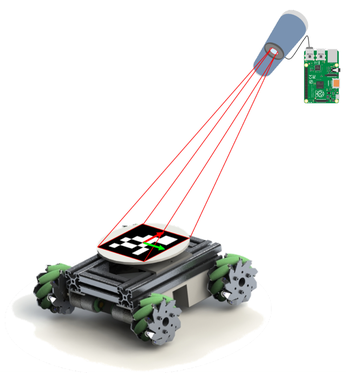
3. HARDWARE COMPONENTS



4. ROBOT COORDINATION

i. Supervisory controller (computer) monitors robot activity through camera vision
ii. Supervisory controller dispatches commands to robotic units wirelessly via Bluetooth Low-Energy (BLE)
iii. Communication controller (Raspberry Pi) wirelessly receives commands and transmits status
iv. Communication controller relays commands to hardware controller (Arduino Uno) through serial bus
v. Hardware controller actuates motor and monitors directional drift and battery status
vi. Hardware controller transmits command responses and fault indications to communication controller

5. ROBOT AND PACKAGE DETECTION



Web Cam

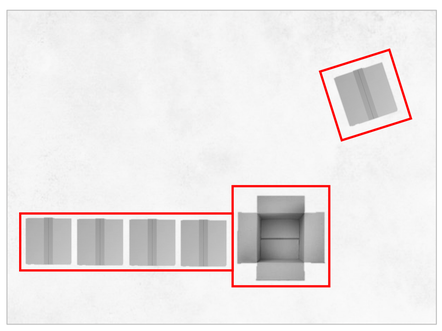
- Live video of workspace processed to identify visual markers
- Printed on rubberized platform of robots
- Via image processing, provides differentiation and pose estimation between robots, outputting 3D direction vector

ArUco Markers

OpenCV

- Computer vision Python library that identifies and processes robot and package locations in real-time via built-in algorithms

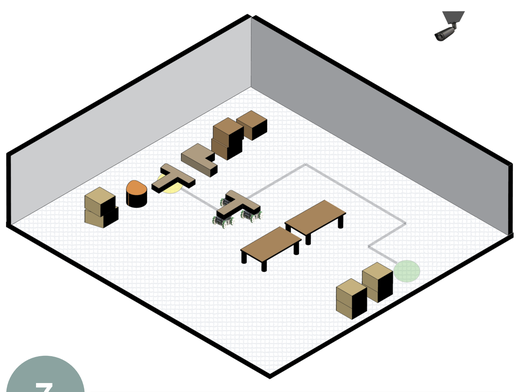
6. OBSTACLE DETECTION



Given a black and white image of the workspace, the obstacle detection algorithm:

- Spikes contrast in image to whiten floor and darken objects
- Determines contrast difference between floor and objects, excluding robots and packages via ArUco markers
- Computes contrasted image histogram, and determines minimum gray-level of objects
- Highlights objects (obstacles) ≥ minimum gray-level

7. PATHFINDING



Given information from previous two steps and destination:

- A* pathfinding algorithm is used to determine shortest distance to specified destination as it is the most ideal algorithm for cases where there is only one source and one destination
- A* computes shortest distance by assigning a cost to each possible route via the cost-heuristic function:

$$f = g + h$$

Where g is cost of moving from a starting point to given cell on grid and h is estimated cost of moving from given cell to destination

- A* picks the path that minimizes the heuristic function

CONCLUSION

The integration of all of these components creates a swarm robotics system that facilitates dynamic delivery within the warehouse space. This reduces the costs and degree of maintenance required to procure and implement individualized robots that perform the same task, thus increasing workflow efficiency via a cohesive design.

NEXT STEPS



Collaborative robot (cobot) design, allowing safe work alongside humans and safe handling practices of the technology by humans



Implementing dynamic pathfinding given moving obstacles (e.g. people, other robots) rather than stopping and waiting until obstacle has cleared



Handling of packages that are irregularly shaped (e.g. have large curves, divots etc.)

University of Guelph

Engineering

Advisor: Dr. Andrew Gadsden, PhD, P.Eng., P.M.P., FASME

Mechanical | Systems and Computing | Computer | Biomedical