

ADAS Forklift

ADAS Systems

- **Collision Avoidance:** the purpose of this feature is as a "supervisor" to the manual driver. Collision avoidance would take in information from a range finding sensor (stereo vision, LIDAR, radar, etc), to predict and prevent collisions from occurring. This feature leaves the driver in control of the situation, but helps augment their ability by providing an additional layer of insulation. This is especially helpful in navigation through blind spots such as close to the front of the forklift or immediately behind; areas where the driver would normally have a difficult time seeing.
- **Autonomous "Parking" for picking up loads:** The purpose of this feature is to help the operator a navigate through a difficult part of the process picking up objects. In particular, this feature is aimed to help align the driver in the correct orientation to pick up a pallet that would have otherwise been tricky to maneuver due to poor visibility.

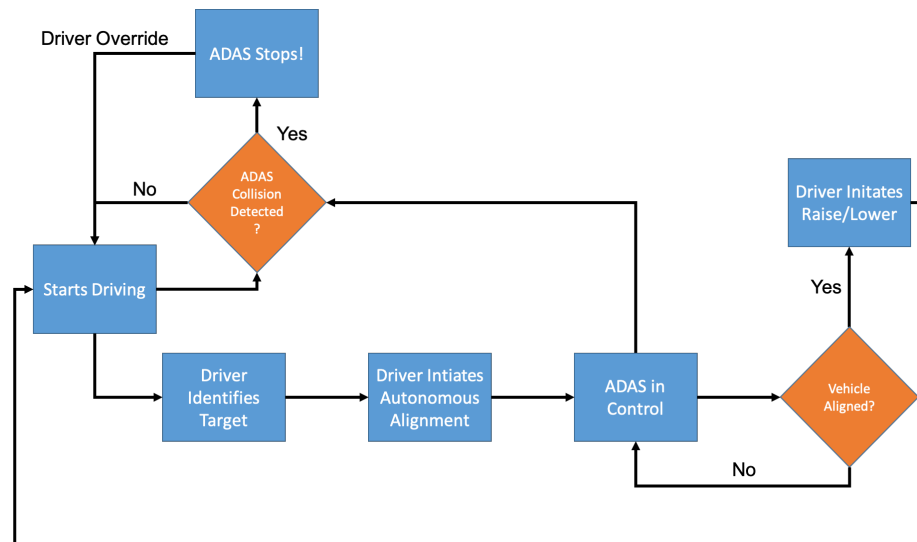
Human Inputs

- Steering
- Accelerate/De-accelerate
- "Select target"
- "Align for pickup"
- "Raise/lower" forklift
- "Manual Override"

Problem Statement

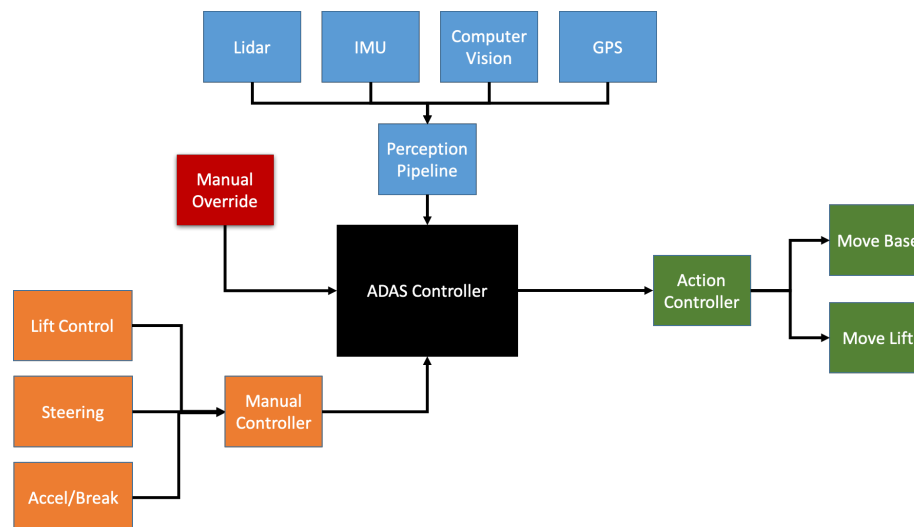
- To develop a system capable of ensuring the saftey of human lives while facilitating fast and efficient picking and placing of industrial scale packages.

Physical Process



Process Flow:

- Driver has manual control (observed by ADAS)
- If ADAS predicts collision, control switches to ADAS to initiate stop
- Driver can select a desired target to pick up and cede control to ADAS
- ADAS will position itself in alignment to pick up package/payload
- During any autonomous position, driver can override back to manual control



Control Overview:

- Sensor data feeds into a perception pipeline
- Manual control feeds into the manual controller
- Perception pipeline, manual controller, manual override feeds into ADAS controller
- ADAS Controller acts as intermediary between manual control and autonomous control. Output control commands are some blend of manual and autonomous.
- Manual override gives complete control to manual controller (disables auton.)

Problems:

- Obstacle Detection/Avoidance
 - Variables:
 - * Static Objects
 - * Dynamic Objects
 - * Noise
 - * Time
 - Inputs:
 - * Camera: object detection
 - * LIDAR: object detection/depth estimation
 - * IMU: time of collision estimation
 - Outputs:
 - * Bounding boxes around obstacles (with depth estimation)
 - * Control functions to avoid collisions
- Target Detection
 - Variables:
 - * Static Objects
 - * Dynamic Objects
 - * Noise
 - Inputs:
 - * Camera: object detection
 - * LIDAR: object detection/depth estimation
 - * User Input: target selection
 - Outputs:
 - * Bounding boxes around obstacles (with depth estimation)
- Allignment
 - Variables:
 - * Static Objects
 - * Dynamic Objects
 - * Noise
 - Inputs:
 - * 3D pose estimates of target

- Outputs:
 - * Bounding boxes around obstacles (with depth estimation)
 - * Control functions for "controlled collisions"

Failure Points:

- Poor lighting: interferes with camera's ability to see/detect objects
- Poor weather (rain/snow/dust/leaves): interferes with LIDAR and camera's ability to see/perceive environment
- Overly crowded environment confuses autonomous systems. May result in standstill overly aggressive avoidance

Optimization:

Optimize Targeting:

- modify environment slightly with some sort of highly reflective markers that are unique and easy to see in camera/LIDAR. Using known markers can help optimize sense of scale.
- restrict operation environment to eliminated "uncontrollables" such as weather (dust, snow, rain) or unexpected dynamic obstacles (sudden cyclists/pedestrians")
- operate in well lit environment