ECE434 Pedestrian Crosswalk Recognition

Team Members: Jacob Yax, KJ Hobday, Tuan Nguyen

Risk Significance

“In 2021 alone pedestrian fatalities in the United States alone numbered 7,388 while over 600,000 were injured, a 13% increase from 2020”

-National Highway Transportation Safety Agency

Mitigations

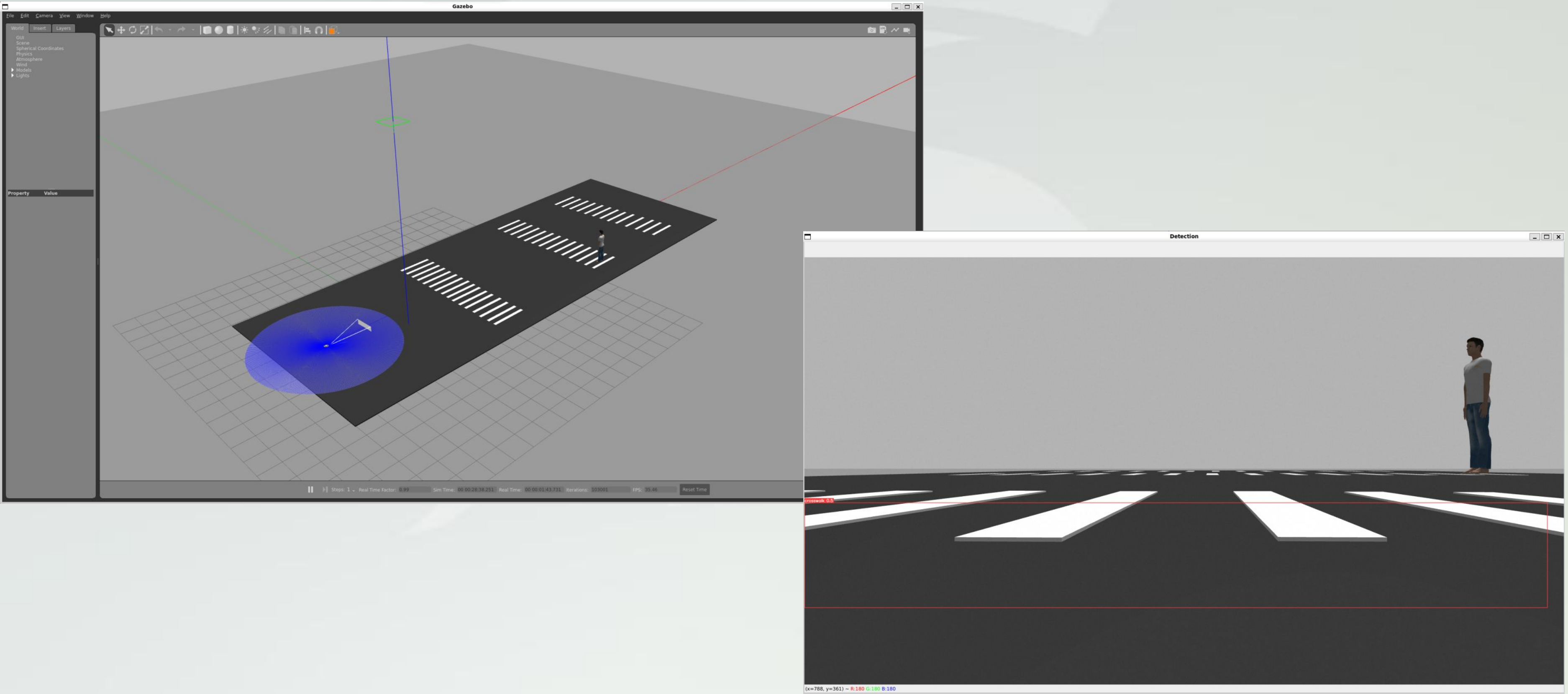
What can an autonomous vehicle do?

- Sense how far along a crosswalk the pedestrian is and control speed based on that
- Fully stop at crosswalk check for pedestrians then move
- Check a pedestrian light to see if red or green and move based off the status of that

Approach

1. Create a world in gazebo to simulate crosswalk environment
2. Utilize images from environment to train robot on what is a crosswalk
3. Create ROS nodes which tell robot when to stop and go as well as determine light
4. Evaluate model to confirm performance

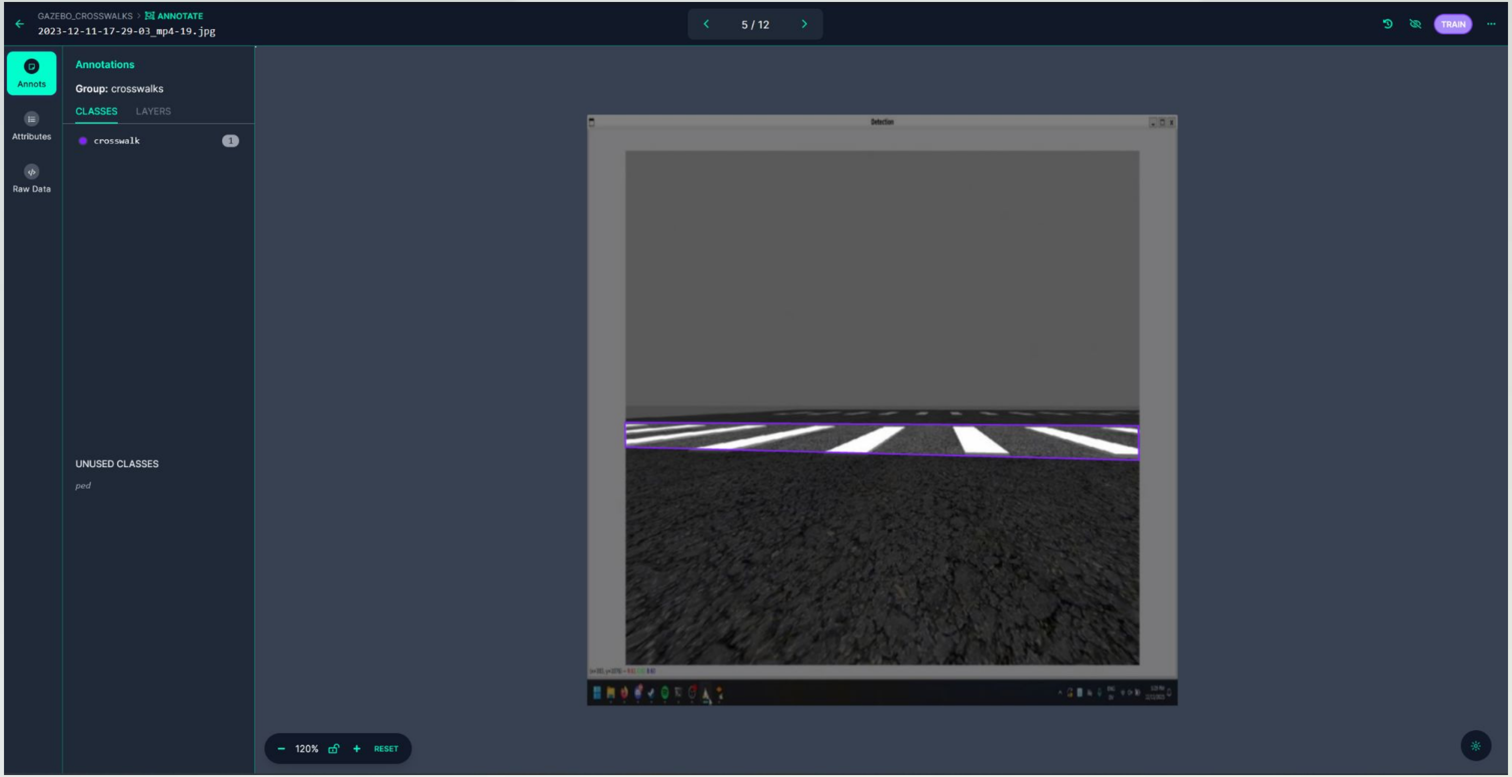
The World



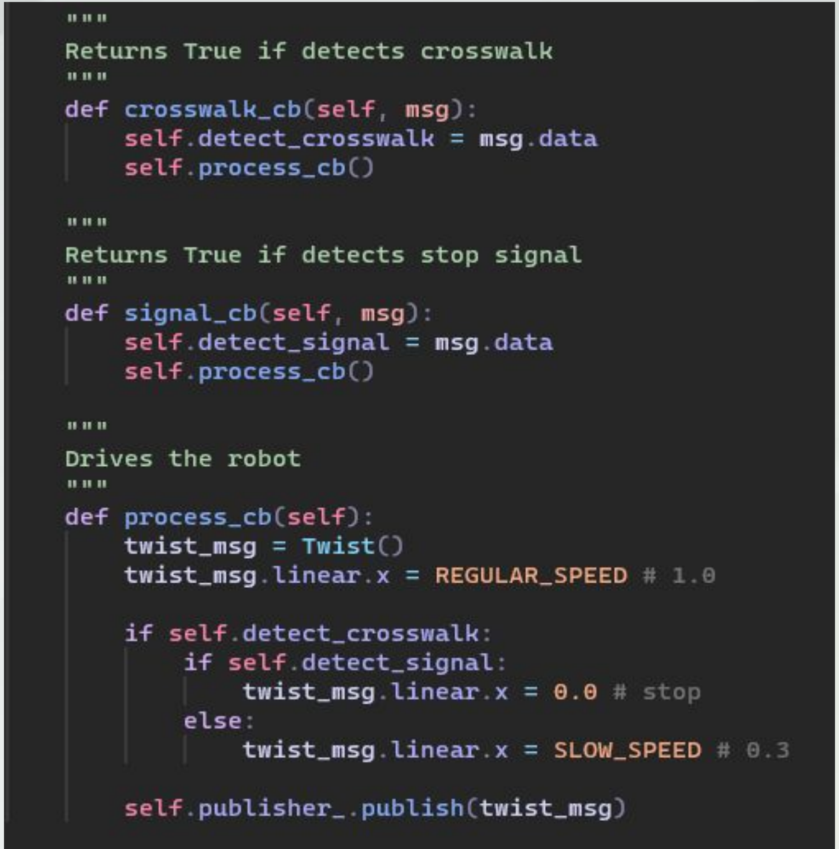
Steps: 1 Real Time Factor: 0.99 Sim Time: 00 00:28:38.251 Real Time: 00 00:01:43.731 Iterations: 103001 FPS: 35.46

(x=788, y=361) ~ R:180 G:180 B:180

Annotating on Roboflow

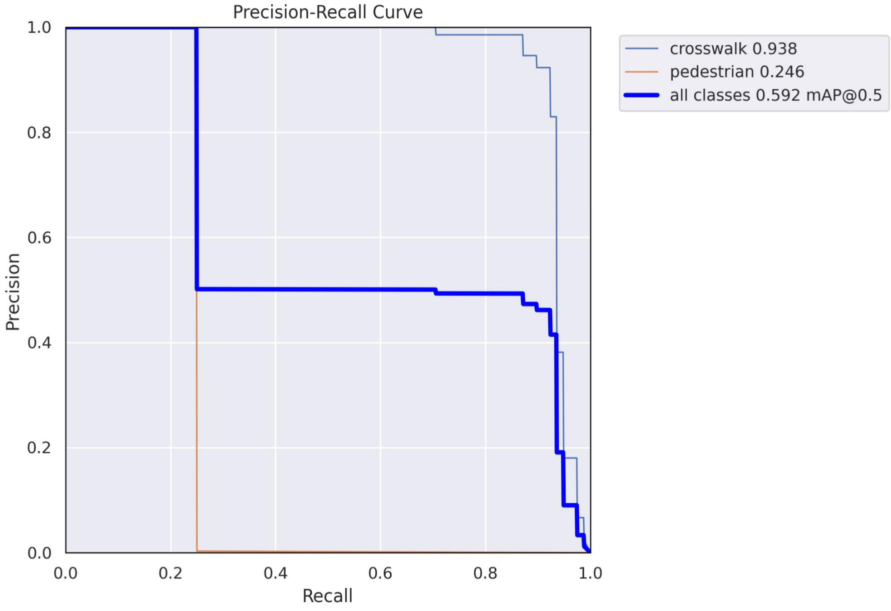


Robot Control Logic

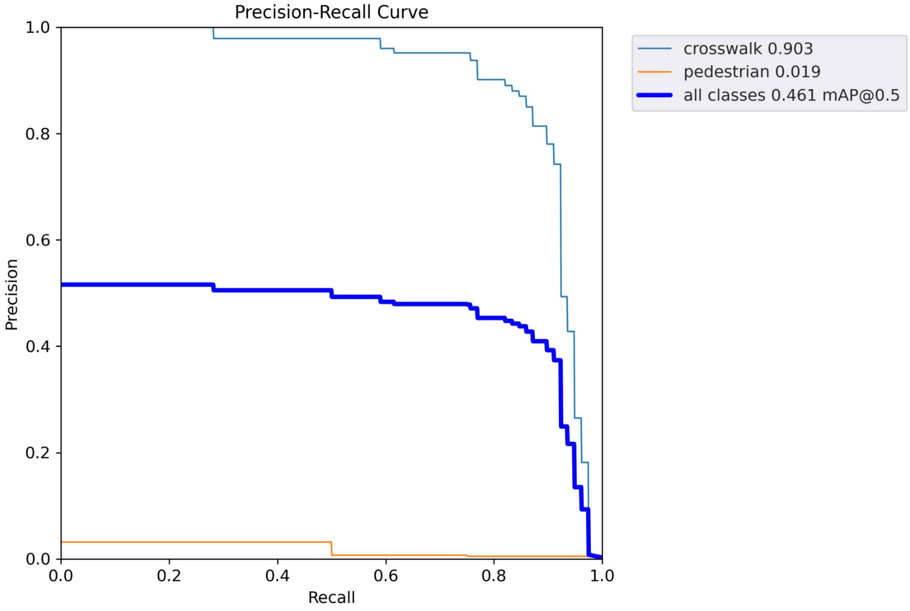


```
"""
Returns True if detects crosswalk
"""
def crosswalk_cb(self, msg):
    self.detect_crosswalk = msg.data
    self.process_cb()

"""
Returns True if detects stop signal
"""
def signal_cb(self, msg):
    self.detect_signal = msg.data
    self.process_cb()

"""
Drives the robot
"""
def process_cb(self):
    twist_msg = Twist()
    twist_msg.linear.x = REGULAR_SPEED # 1.0

    if self.detect_crosswalk:
        if self.detect_signal:
            twist_msg.linear.x = 0.0 # stop
        else:
            twist_msg.linear.x = SLOW_SPEED # 0.3

    self.publisher_.publish(twist_msg)
```

Training Sets



Real World



Virtual World

Results

Conclusion

In a world where safety remains the top concern with autonomous vehicles, this model offers a successful answer on handling pedestrian safety