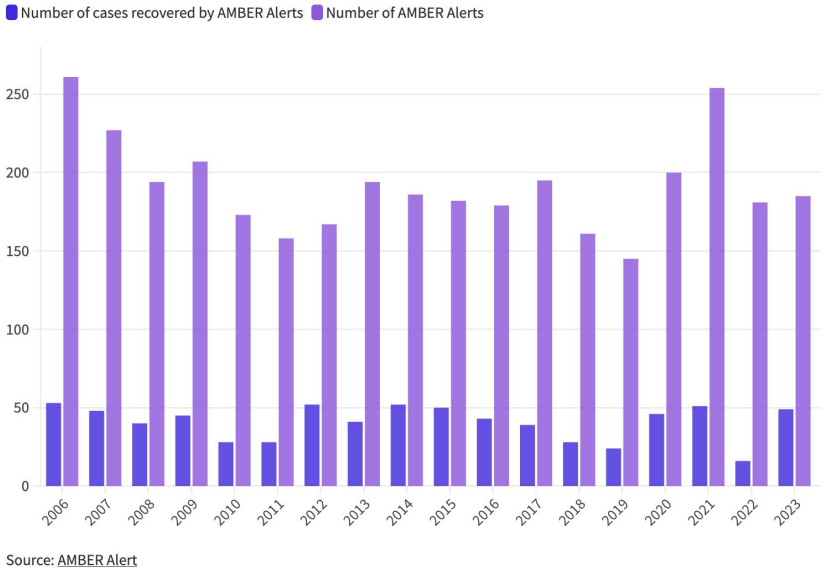
Background

- In 2023, 185 AMBER Alerts were issued in the U.S. involving 229 children
  - 26% were resolved as a direct result of Amber Alerts
  - Only 8% of cases in 2022
- 840,000 children go missing per year
- Amber Alerts are sparsely used
  - In 2019, there were more than 3,500 cases of missing children in the state of South Carolina last year, yet there was only one AMBER Alert.

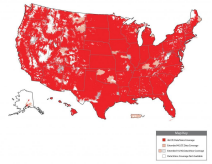


Source: AMBER Alert

1 *Sources:*
https://www.newsweek.com/chart-shows-how-successful-amber-alerts-are-recovering-missing-children-1952170
https://www.nbcdfw.com/news/national-international/do-amber-alerts-really-bring-missing-children-home/3620115/
https://www.cnn.com/2020/02/22/us/amber-alert-explainer/index.html

Use Case

- Most people do not actively look for amber alerted vehicles
- Our device will get more people participating while not distracting drivers
  - Camera, microprocessor, cloud server/database system
  - Constant automatic detection
  - Common weather and lighting conditions (rain, snow, nighttime, etc.)
- By getting more eyes in the area, we can greatly increase the contribution of Amber Alerts to recovering missing children
- Hardware and software areas



Companies in the Space

Existing solutions often target law enforcement or companies, and are expensive:

- PLATESMART MOBILE DEFENDER
- Plate recognizer
- Genetec AutoVu

We will differentiate with price, accessibility, and target users

- Low-cost and easy to install
- General public as users
- Used for amber alerts specifically

Sources: https://www.platesmart.com/platesmart-mobile-defender/
https://platerecognizer.com/
https://www.genetec.com/products/unified-security/autovu

A Note On Privacy

We acknowledge that privacy is a large concern when it comes to using license plates to track a car's location. We will address these concerns by:

- Edge computed detection
- Easy Opt In / Opt Out
- Anonymous sender

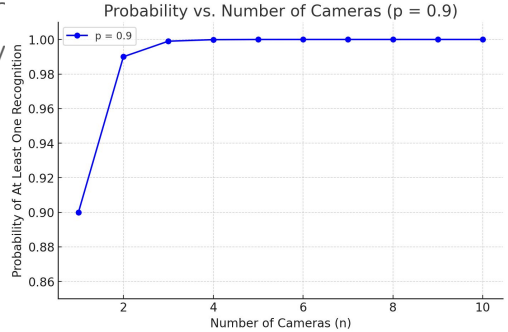
Concerns out of the scope of the project:

- Malicious attacks
- False amber alerts
- Modified license plates

Use Case Requirements

License Plate Recognition Accuracy: The system must achieve a license plate recognition precision of at least 90% to reduce noise and 85 % recall in common driving lightning and weather conditions

- A previous study shows that 90% precision is achievable for plate recognition and we would want the highest possible
- We want to focus more on limiting false positives th false negatives since false positives are more costly a police department



License Plate Detection Range: 50 meters minimum

- This is a typical following distance between cars on the highway, and would be a common distance between the dash cam and license plates.

Source: Enhanced Vehicle Identification: A Machine Learning Approach to Number Plate Recognition https://ijrpr.com/uploads/V5ISSUE6/IJRPR29970.pdf

Use Case Requirements

Processing Time: less than 60 seconds

- This is so captures can be taken and processed in regular, frequent intervals to reflect the constantly changing nature of the cars around the user, but also gives time for the slower processing of edge computation.

Ease of integration:

- The dash cam system should be easy to install, relatively compact, and not block the driver's view to minimize downsides to the drivers who are needed for the system to work.

Technical Challenges

- Exploring viability of retaining high recognition accuracy and speed while doing edge compute
- Ensuring accuracy under various lighting / noise / weather / speed conditions
- Operating under the capabilities of our edge computing devices
  - Running high-intensity inference models on small devices may cause overheating
- Making sure the cloud server has a timely response at all times
- Making sure irrelevant text near a license plate is not detected as one
- If edge compute requirements are infeasible, we will move the ML model to the cloud

Solution Approach

- Camera setup (Raspberry Pi camera-module-3)
  - Collects visual information and forwards it to microprocessor with GPIO
  - Mounted to the inside of the car facing out of the front windshield
  - Inexpensive, IR capabilities for low lighting, autofocus, easy to integrate with raspberry pi
- Microprocessor (Raspberry Pi)
  - Runs an ML model for character or car recognition and forwards information to cloud
    - Using a CNN for locating and cropping licence plates from images and a character recognition model for processing it into recognized letters
  - Low power, small form factor, inexpensive, easy to integrate
- Cloud
  - Matches information from the microprocessor with an SQL database of amber alerts and relevant information
  - Depending on our specific direction, may run larger models like vision transformers for more difficult recognition tasks
  - Inexpensive computation and data storage, scalable

Testing, Verification, and Metrics

Compute testing:

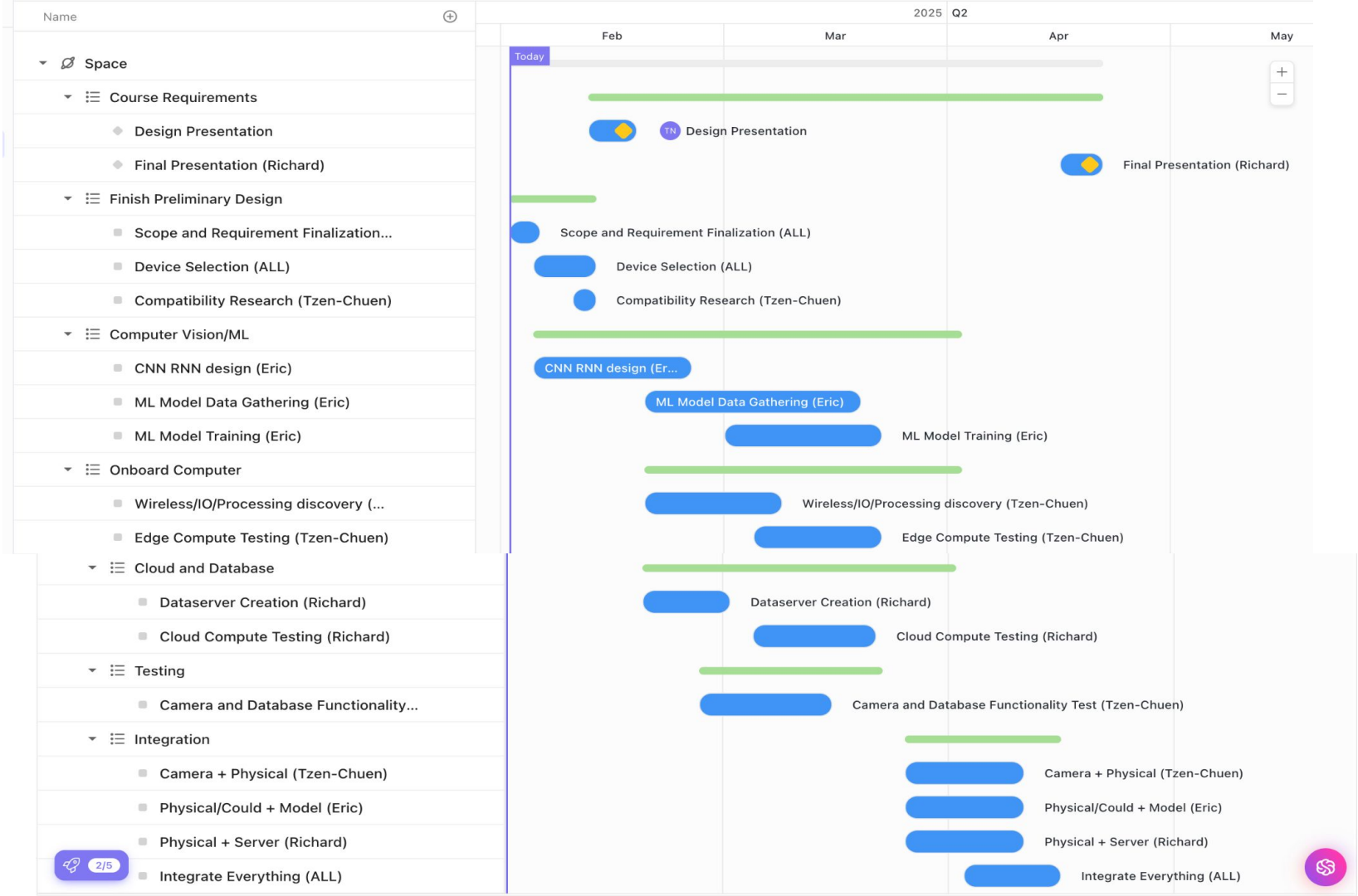
- Performance of Edge vs. Cloud Compute
  - Recognition rate
  - Size of Model
  - Time to recognition

Ease of integration testing:

- Can be installed and setup within 15 minutes in a car

Unit testing:

- Recognition accuracy:
  - Test using a fixed setup varying conditions like distance, lighting, decals, etc
  - Measuring precision and recall over the set
- Integration testing:
  - The whole system is able to take in the license plate and vehicle image data to generate the corresponding plate number within 60 seconds


Name
2025 Q2
Feb
Mar
Apr
May
Today
Space
Course Requirements
Design Presentation
Final Presentation (Richard)
Finish Preliminary Design
Scope and Requirement Finalization...
Device Selection (ALL)
Compatibility Research (Tzen-Chuen)
Computer Vision/ML
CNN RNN design (Eric)
ML Model Data Gathering (Eric)
ML Model Training (Eric)
Onboard Computer
Wireless/IO/Processing discovery (...
Edge Compute Testing (Tzen-Chuen)
Cloud and Database
Dataserver Creation (Richard)
Cloud Compute Testing (Richard)
Testing
Camera and Database Functionality...
Integration
Camera + Physical (Tzen-Chuen)
Physical/Could + Model (Eric)
Physical + Server (Richard)
Integrate Everything (ALL)
TN
Design Presentation
Final Presentation (Richard)
Scope and Requirement Finalization (ALL)
Device Selection (ALL)
Compatibility Research (Tzen-Chuen)
CNN RNN design (Er...
ML Model Data Gathering (Eric)
ML Model Training (Eric)
Wireless/IO/Processing discovery (Tzen-Chuen)
Edge Compute Testing (Tzen-Chuen)
Dataserver Creation (Richard)
Cloud Compute Testing (Richard)
Camera and Database Functionality Test (Tzen-Chuen)
Camera + Physical (Tzen-Chuen)
Physical/Could + Model (Eric)
Physical + Server (Richard)
Integrate Everything (ALL)
2/5

Division of Labor

Richard:

- Data Server Creation
- ML integration with chip or cloud
- Cloud database testing

Eric:

- Initial ML implementation
- Research / test viability of more recent character recognition architectures
- Accuracy testing

Tzen-Chuen

- Edge computing
  - Wireless Data Transfer
  - Geolocation
- Camera Data Processing