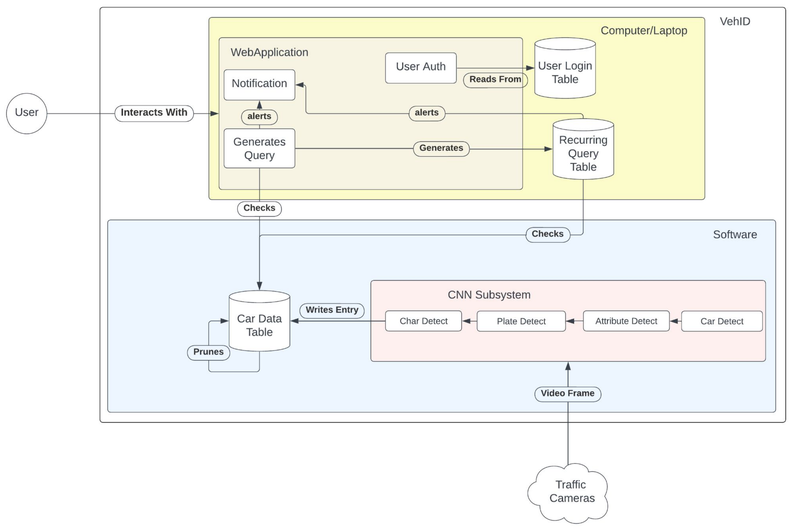
# VehID

Remington Greko, Spencer Hirsch, Thomas Johnson, Alexis Nagle
Faculty Advisor(s): Marius Silaghi, Dept. of Computer Science, Florida Institute of Technology

## Goal

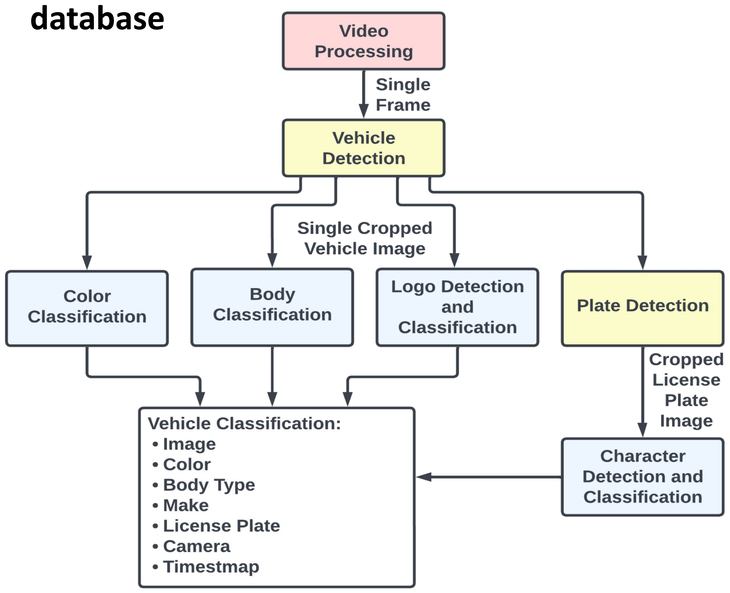
To improve public safety by utilizing machine learning to recognize vehicles based upon a variety of characteristics such as color, body type, make, and/or license plate.

## Design




## Neural Network Subsystem

- Utilized pre-exisitng NN architectures
  - MiniVGG (Body and Color Classifications)
  - VoloV8 (Vehicle, Logo, PLate, and Character Detections)
- Exports results to a JSON file for use in the database




## Motivation

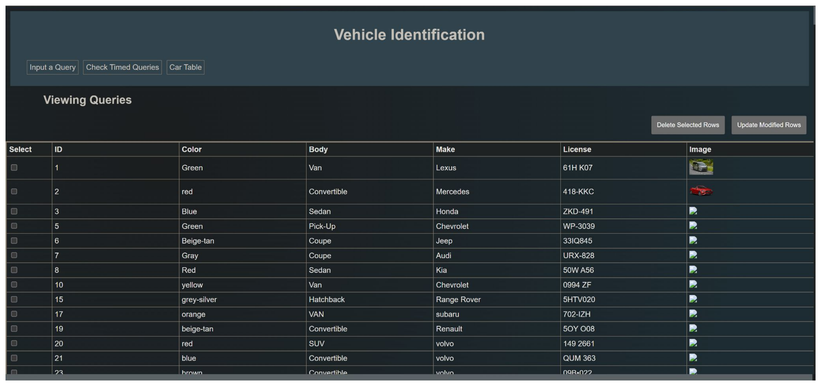
- Aid in AMBER alerts, stolen vehicles, and criminal offenses, which tends to rely on pure human interaction to spot and report the specified vehicles.
- Existing automation only identifies license plate numbers and is not beneficial when only other characteristics are available.

## Database

- Car database: Stores entries of vehicles detected by the NN subsystem.
  - Includes image path, color, body type, make, plate, camera ID, and timestamp.
- Query database: Contains user generated queries to search for full/partial matches
  - Includes query parameters and refresh period
  - Executes each query by searching for the defined characteristics at the user defined refresh period

## Web Application

- Allows for users to interact with the data contained within the databases.
- User functionalities:
  - Create queries to search for desired vehicles
  - Filter queries and car entries to be displayed on their respective web pages
  - Edit and delete queries or car entries
  - View saved images for each car entry for manual verification




## Implementation

- Utilized various Python libraries
  - OpenCV
  - Tensorflow: Keras, MiniVGG model
  - Ultralytics: YoloV8 models
- Web application tools
  - html, css, js
  - node

## Evaluation Results

## Limitations

- Image Quality
- Weather/Environmental Factors

## Improvements

- Further training and tuning on select models to improve individual performances.
- Implement web scraping to automatically populate the database with queries from sources such as AMBER alerts.
- Incorporate into a network of existing cameras.