Master in
Computer Vision
Barcelona

Module: C6 - Project Presentation Part 1

Project: Visual Recognition

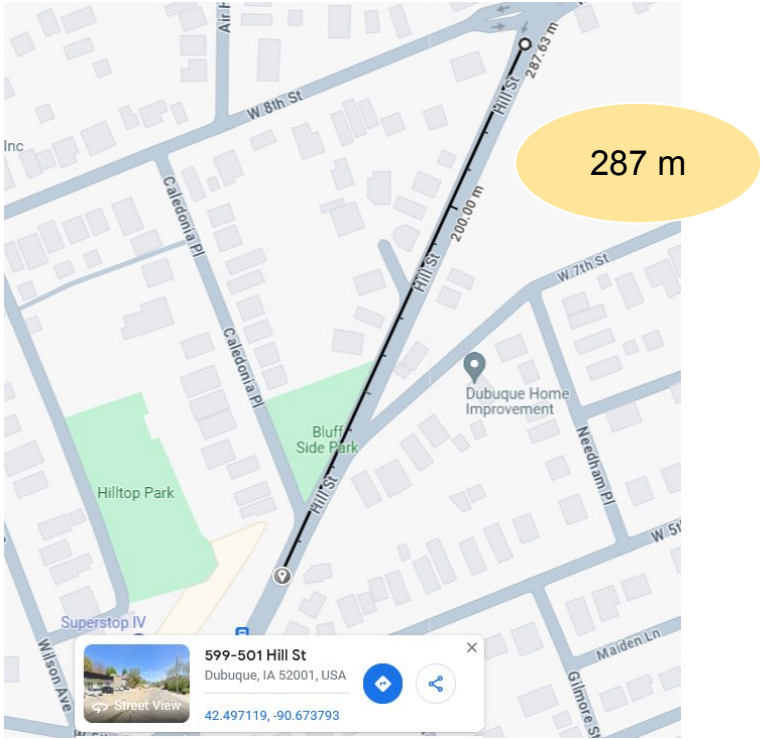
Team 2: Miruna-Diana Jarda, Gunjan Paul, Beatrice Anamaria Peptenaru, Diana Tat

Task 1.1: Speed estimation

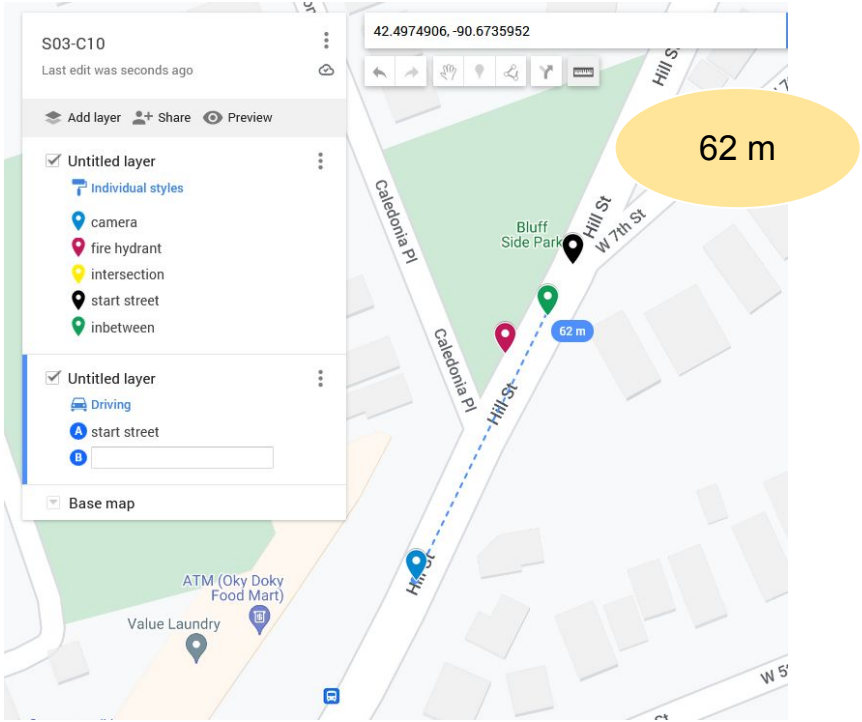
Diana Tat 1689764

Diana Tat 1689764

Task 1.1: Speed estimation



Whole street (from camera to faraway intersection)

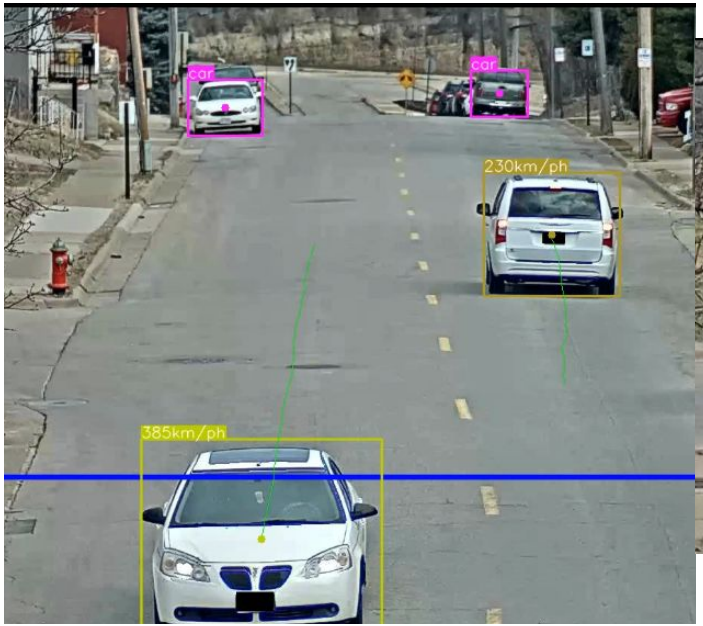


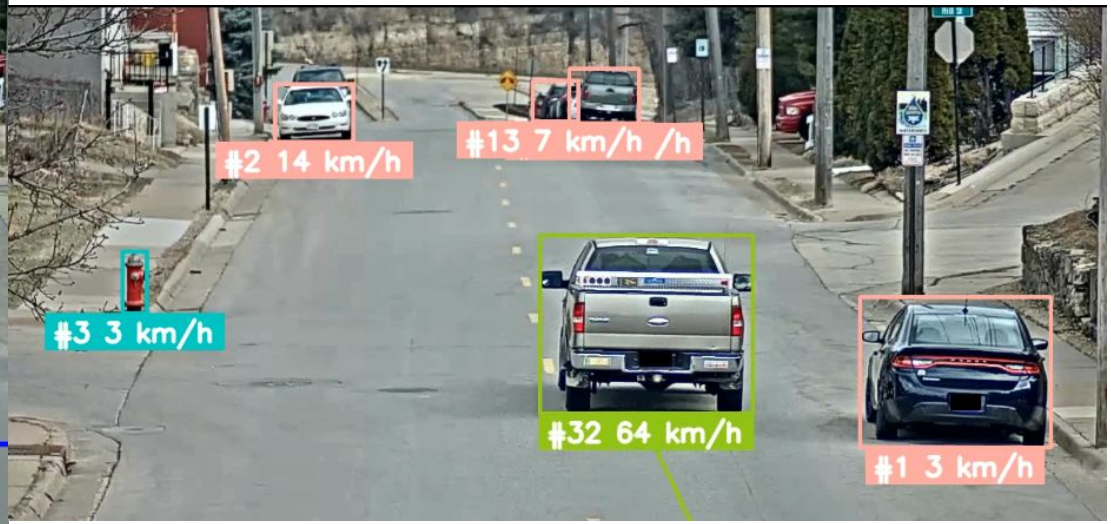
Region of interest (given by the 2 lines)

Diana Tat 1689764

Task 1.1: Speed estimation

Trying to find a ground truth or at least a starting point - harder than





Both methods are using YOLO, first one is using the YOLO speed estimator, second is using a method from a RoboFlow tutorial

Task 1.1: Speed estimation

Diana Tat 1689764

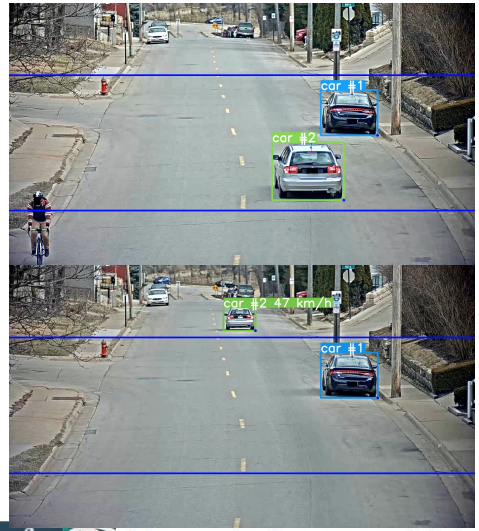
Starting frame

+

Mask (ROI for detection)

Car is inside

Car is outside but has previously been inside



Increase time

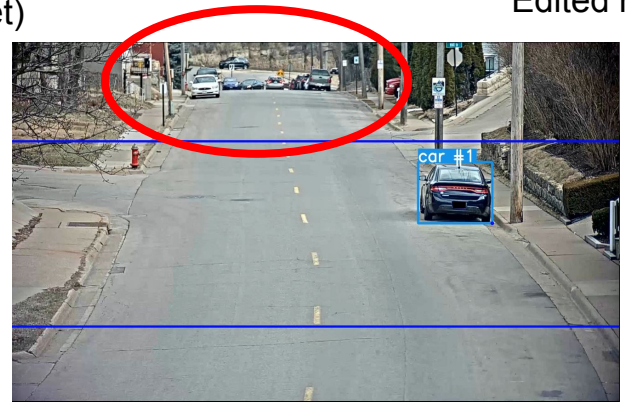
Calculate speed

Diana Tat 1689764

Task 1.1: Speed estimation

Original mask (from dataset)

Edited mask



Result - ignoring far away cars

Task 1.1: Speed estimation

Diana Tat 1689764

Task 1.1: Speed estimation (seq3-c010)

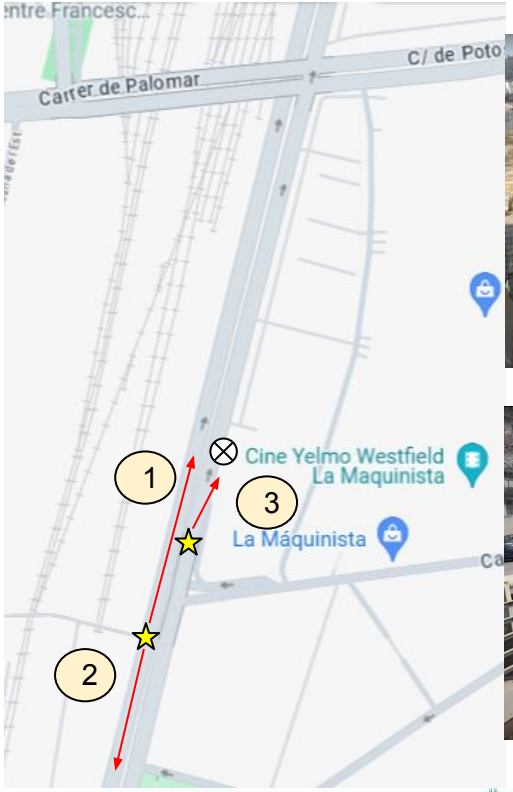
Diana Tat 1689764

	Estimated speed (km/h)
Vehicle 2	47.48
Vehicle 4	37.2
Vehicle 5	41.33
Vehicle 6	34.33
Vehicle 9	31.0
Vehicle 10	51.90

	Estimated speed (km/h)
Vehicle 12	35.42
Vehicle 13	43.76
Vehicle 14	43.76
Vehicle 15	29.36
Vehicle 17	43.76
Vehicle 18	42.11

Beatrice Peptenaru 1616043

Task 1.2: Use your own data



View 1 - referred to as ‘zoom’ (zoom 1.3x)
View 2 - referred to as ‘tunnel’
View 3 - referred to as ’radar’ (wide angle 0.5x)

Camera: Iphone 11
Resolution: 1920x1080
FPS: 30, reduced to 10

★= camera position

Beatrice Peptenaru 1616043

Task 1.2: Use your own data

🚫 Cons :

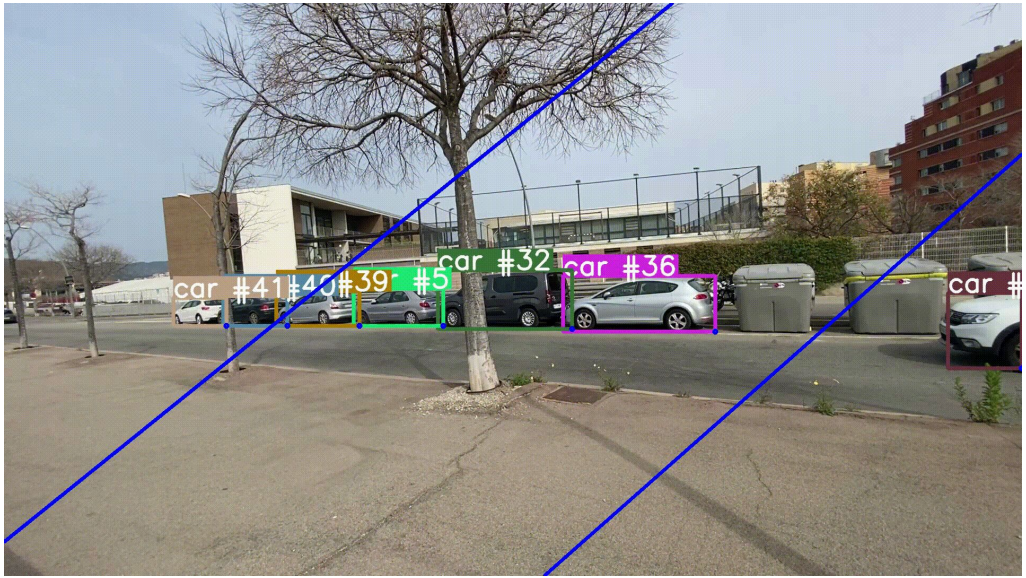
- Small distances to measure speed in
- Vanishing point very far away
  (hard to see labels in some cases)
- Tunnels (car outside region for measuring / taking too much space from potential detection region)

✚ Pros:

- Easy to delimitate region on interest for detection and speed measurement
- No parked cars on the main road / easy to ignore parked cars

Beatrice Peptenaru 1616043

Task 1.2: Use your own data



🚫 Cons :

- Perspective (harder to delimitate region on interest for detection and speed measurement)
- Parked cars keep changing the id after being obscured (=> hard to remove them)

➕ Pros:

- Radar for ground truth

Task 1.2: Use your own data (tunnel vs zoom)

Beatrice Peptenaru 1616043

Tunnel	Estimated speed (km/h)
Vehicle 2	64.8
Vehicle 5	36.0
Vehicle 6	54.0
Vehicle 8	49.84
Vehicle 9	46.28
Vehicle 10	46.28
Vehicle 11	64.8
Vehicle 13	49.84
Vehicle 15	64.8
Vehicle 16	64.8

Tunnel	Estimated speed (km/h)
Vehicle 21	58.90
Vehicle 23	64.8
Vehicle 29	81.0
Vehicle 30	72.0
Vehicle 34	64.8
Vehicle 40	54.0
Vehicle 42	58.90
Vehicle 43	46.28
Vehicle 44	54.0

Zoom	Estimated speed (km/h)
Vehicle 2	37.38
Vehicle 3	30.37
Vehicle 4	40.5
Vehicle 5	40.5
Vehicle 6	40.5
Vehicle 7	48.6
Vehicle 8	48.6
Vehicle 9	43.76
Vehicle 20	44.18

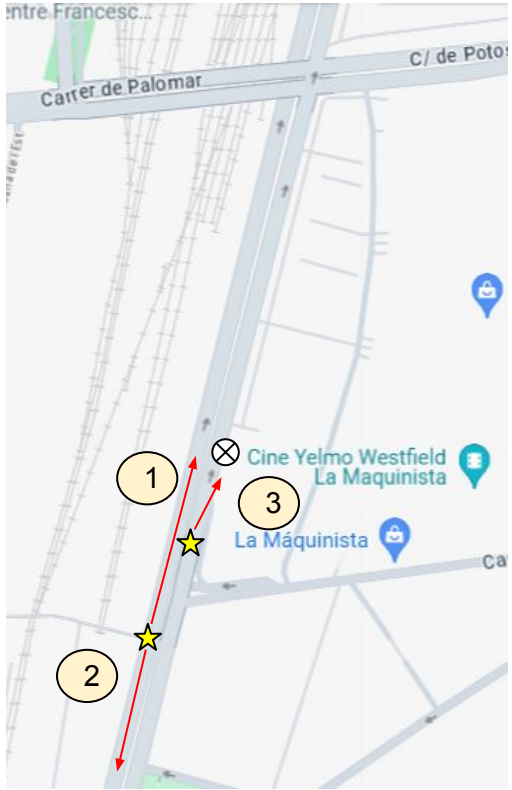
Zoom	Estimated speed (km/h)
Vehicle 21	40.5
Vehicle 22	44.18
Vehicle 25	48.6
Vehicle 27	54.0
Vehicle 28	49.84
Vehicle 29	54.0
Vehicle 30	44.18
Vehicle 35	54.0

Task 1.2: Use your own data

Beatrice Peptenaru 1616043

Tunnel: 13/19 cars going over 50 km/h
Zoom: 3/17 cars going over 50 km/h
< 5km/h over 50

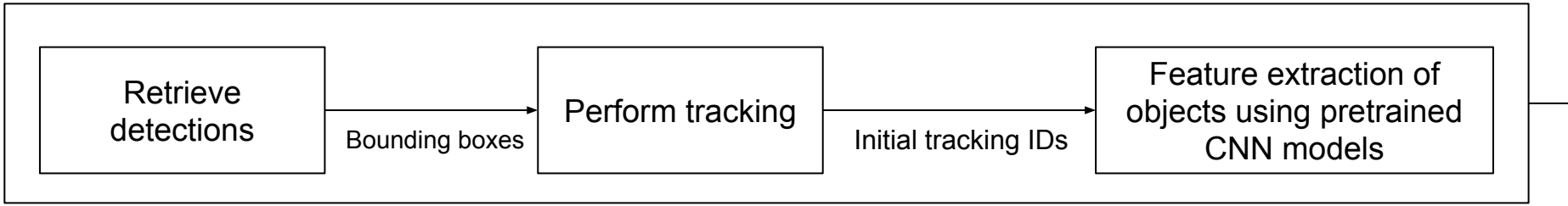
Why?
There is a speed sign on the zoom side!



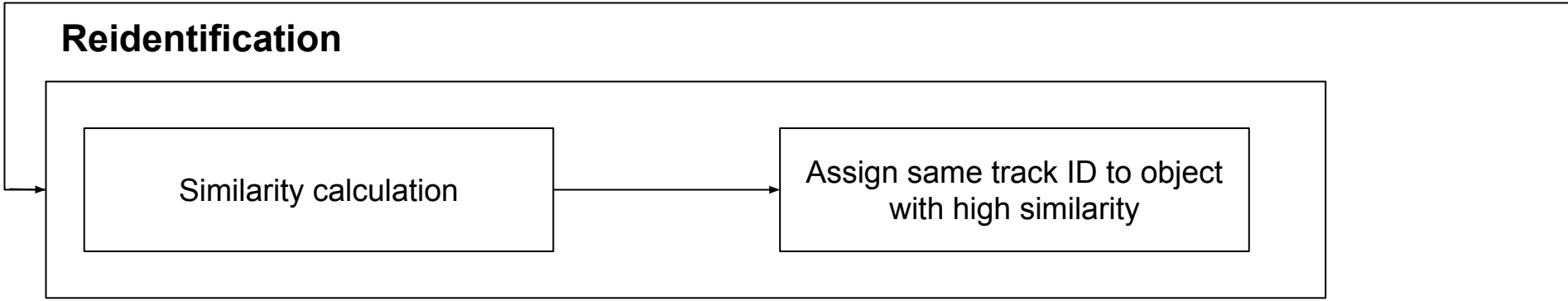
Miruna Jarda

Task 2: Multi camera tracking - Pipeline

Database creation



Reidentification



Task 2: Multi camera tracking - Database creation

1. Retrieve detections
   - We have used the yolov3 detections available in the directory
2. Perform tracking
   - Based on SORT [1]
3. Feature extraction of objects using pretrained CNN models
   - Model used is ResNet50 [2]

[1] A. Bewley, Z. Ge, L. Ott, F. Ramos and B. Upcroft, "Simple online and realtime tracking," *2016 IEEE International Conference on Image Processing (ICIP)*, Phoenix, AZ, USA, 2016, pp. 3464-3468, doi: 10.1109/ICIP.2016.7533003.
[2] K. He, X. Zhang, S. Ren and J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Las Vegas, NV, USA, 2016, pp. 770-778, doi: 10.1109/CVPR.2016.90.

Task 2: Multi camera tracking - Reidentification

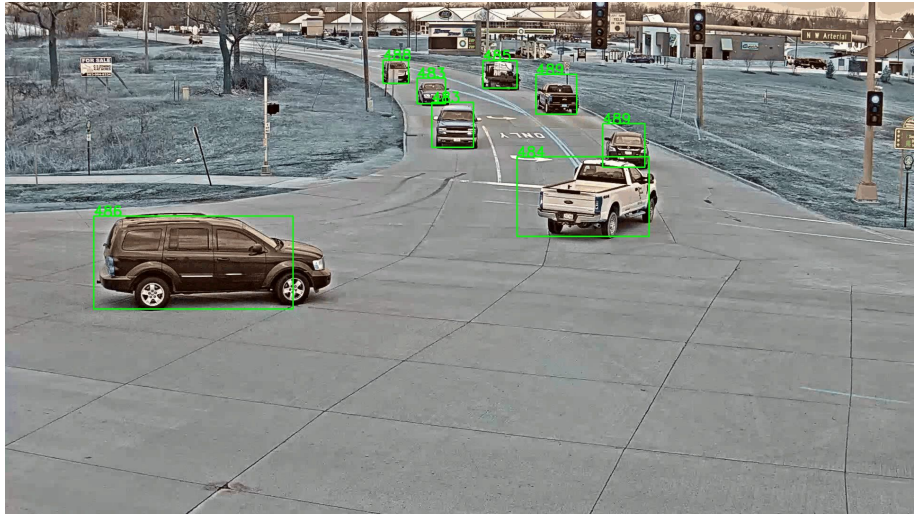
1. Similarity calculation
   - Using cosine similarity
2. Assign same track ID to object with high similarity
   - Consider the track ID of the car with the highest similarity score if above threshold
   - Choose from the track IDs in the other cameras within the sequence
   - Choose within a certain timeframe

Task 2: Multi camera tracking - Assumptions

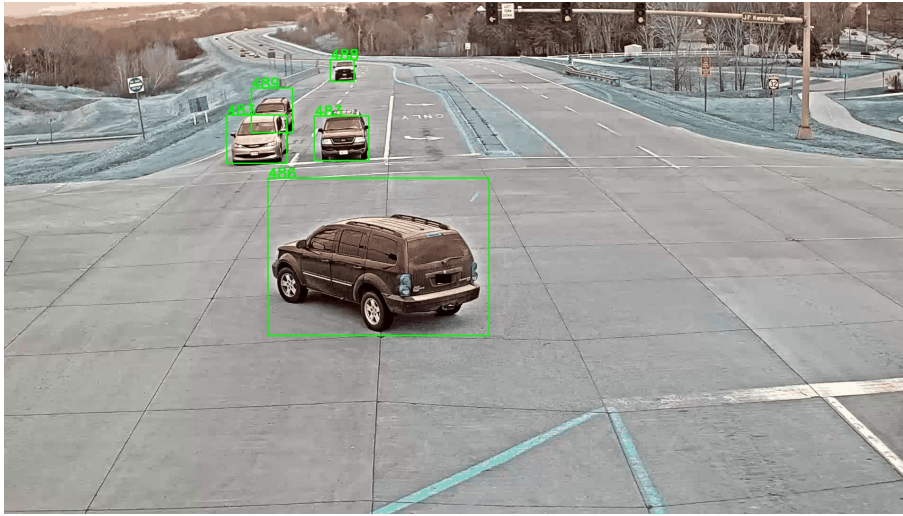
In the creation of our algorithm, we have made two important assumptions:

1. Tracking within one camera works perfectly
2. A car seen by a camera in a frame should be visible to another camera within a certain timeframe
3. All cars are similar to each other to a certain degree
4. Mapping of tracking IDs is transitive

Task 2: Multi camera tracking - Results



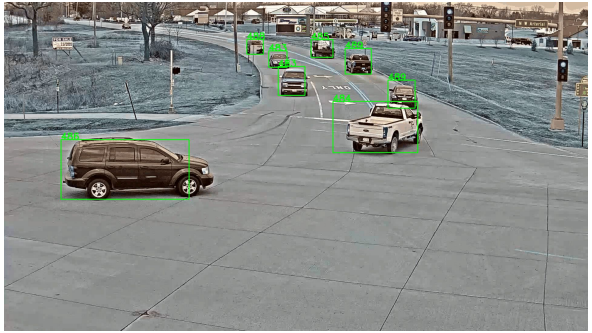
S01/c002



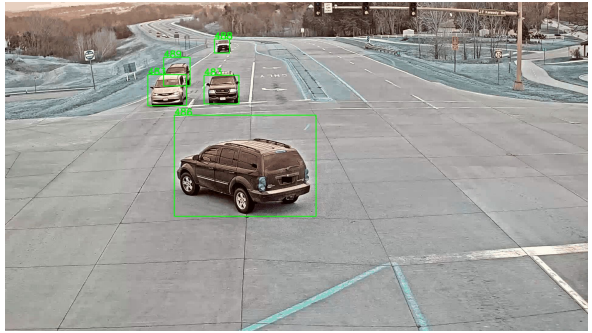
S01/c003

Miruna Jarda

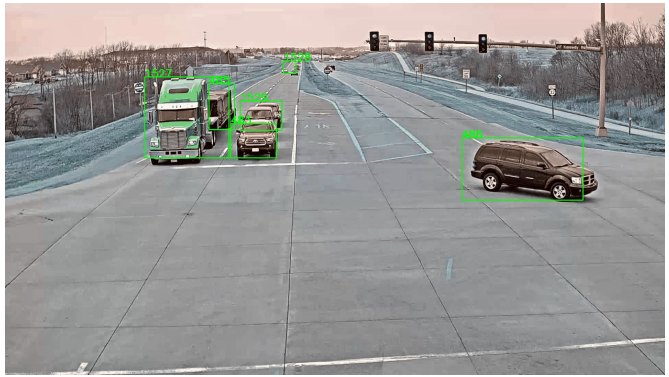
Task 2: Multi camera tracking - Results



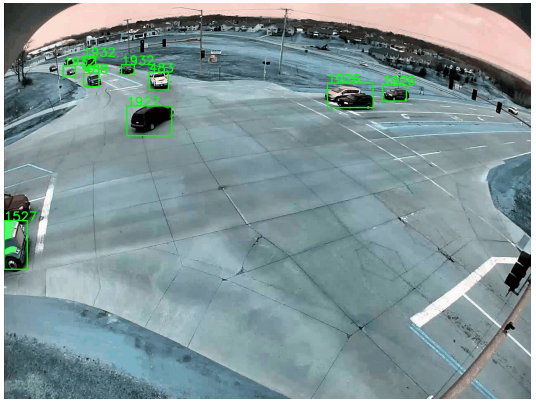
S01/c002



S01/c003



S01/c004



S01/c005

Task 2: Multi camera tracking - Further improvement

The algorithm could be further improved by

1. Optimizing the parameters
    - Trying different models for feature extraction
    - Trying different ways to calculate the similarity
    - Timeframe
    - Threshold
    - …

2. Taking the time offset into consideration for each of the
3. Making predictions of where a car could be at a certain time frame in a certain camera
4. Choosing better feature vectors for the objects

Conclusions

The view (angle + elements of the frame) matter as much, if not more, as the algorithm used to estimate the speed.

- Our multi-camera tracking system has achieved success in identifying some cars across multiple cameras.
- Challenges include handling occlusions, small object appearances, and variations in perspectives among different cameras.
- Implemented as an offline system, allowing for improvements by incorporating additional factors beyond visual data, such as camera matrices or actual car tracking.
- In real-time systems, accessing comprehensive information from the outset poses greater difficulty, potentially limiting optimization possibilities.