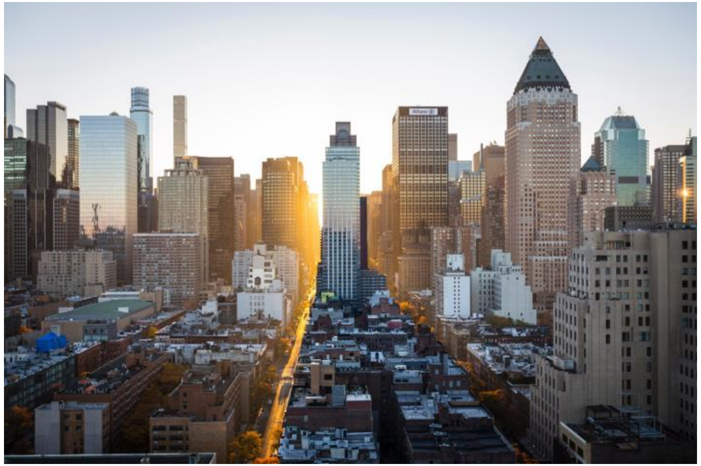


Image Source: Unsplash

Smart Intersection

Open House Presentation
2020 WINLAB Summer Internship

Nicholas Meegan
Kevin Zhang
Bryan Zhu

http://www.winlab.rutgers.edu/ https://bzz3ru.wixsite.com/smartintersection

Team Members

Nicholas Meegan

Rutgers University - Class of 2021

Computer Engineering (major)
Computer Science (minor)

Kevin Zhang

Rutgers University - Class of 2022

Computer Engineering (major)
Statistics (minor)

Bryan Zhu

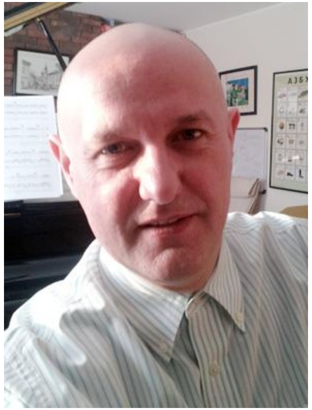
Rutgers University - Class of 2021

Electrical Engineering (major)
Computer Science (major)

Project Advisors

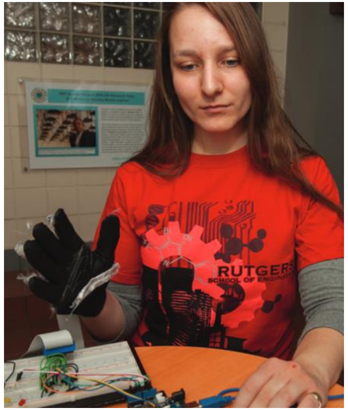
Ivan Seskar

Associate Director
at WINLAB



Zoran Kostic

Associate Professor
at Columbia University



Jennifer Shane

Laboratory Engineer
at WINLAB

Project Overview

About

Smart Intersection **detects and tracks vehicles** as they move through a traffic intersection (a process known as inference), then **analyzes those detections to count vehicles** during different times of day over different days of the week.

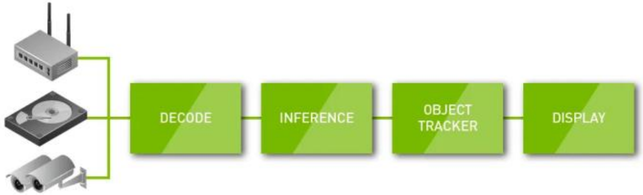
The project aims to aid cities by using its results to offer insights into how cities can best manage their vehicular traffic.

Project Goals

- Detect and track vehicles on traffic intersection footage
- Create system that lets us detect and track vehicles on one computer and view the results on a different computer
- Generate traffic flow statistics using the results from our system

Technologies Employed

NVIDIA DeepStream



a software framework tailored towards video analytics

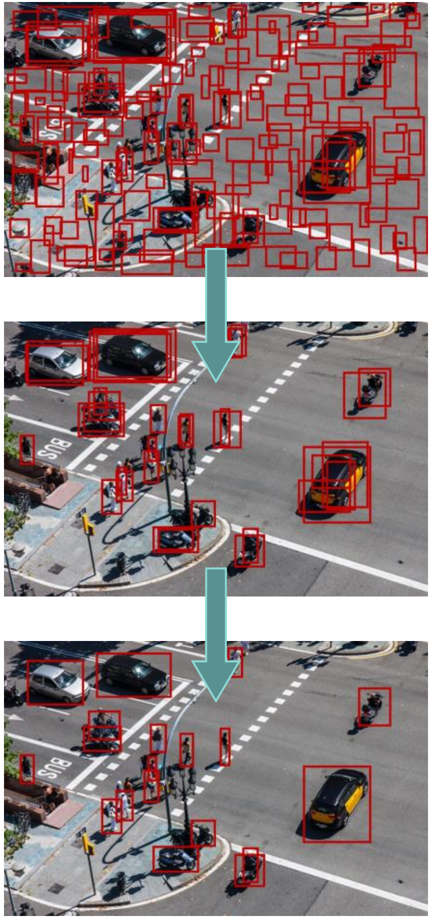
- Leverages graphics hardware for performance increases
  - Optimizes code that runs inference on video data
  - Executes algorithms efficiently
- Streamlines implementation of object detection models

Image Source: http://www.rastek.com/2018/06/accelerate-video-analytics-development-with-deepstream-2-0/

YOLOv3 (You Only Look Once)

the 3rd iteration of a notable real-time object detector

- Classifies objects in a single sweep of the image
- Considers objects in the context of the whole scene
- Performs well even when detecting new versions of known objects



Unedited Image Source: https://www.pxfuel.com/en/free-photo-omeaz

Other Technologies

AMQP 0-9-1 (Advanced Message Queuing Protocol)

Lets us send data between machines (publisher → broker → subscriber)

OpenCV (Open Source Computer Vision Library)

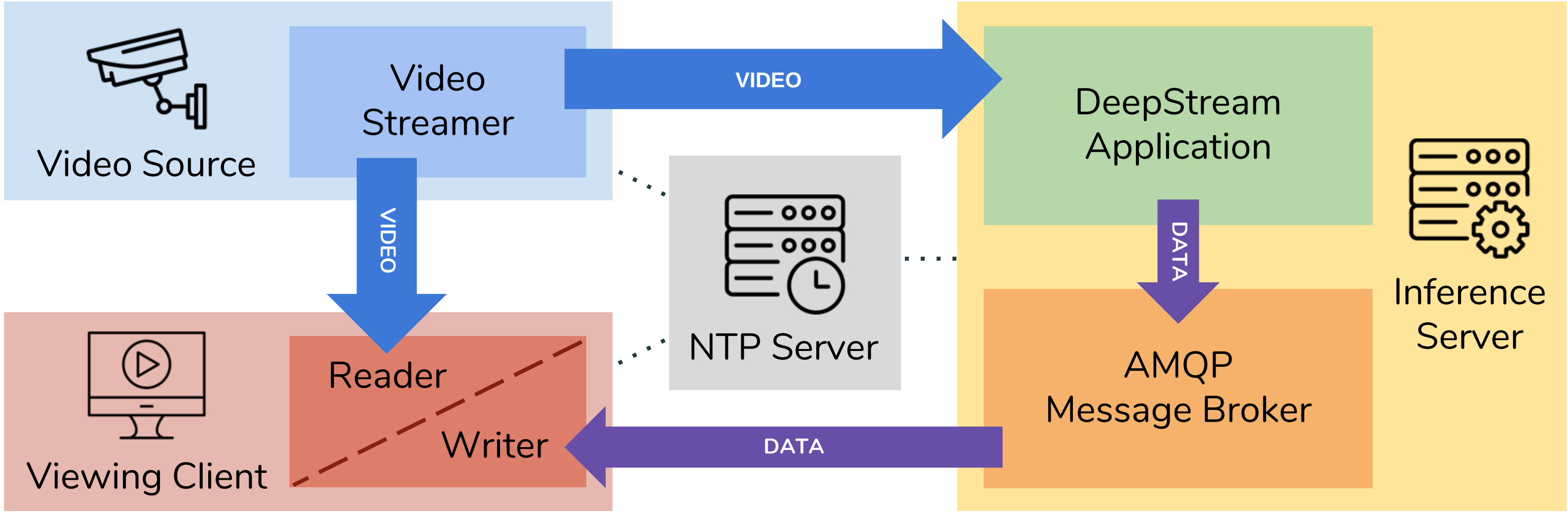
Labels and draws boxes around objects in original video stream

NTP (Network Time Protocol)

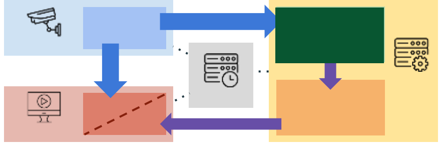
Synchronizes system clocks of all machines involved with project

Implementation

Network Data Flow Diagram



Icon Sources: Atif Arshad (The Noun Project)



DeepStream Application Pipeline

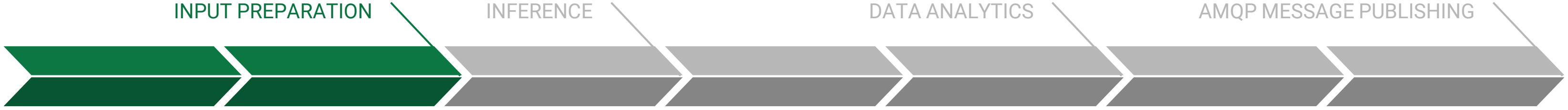
- The application receives a video stream from a source
    - Individual video files
    - RTSP streams (live streaming video)
- The multiplexer adjusts the resolution of the stream for inference
- Timestamps are taken for each video frame

INPUT PREPARATION

INFERENCE

DATA ANALYTICS

AMQP MESSAGE PUBLISHING



URI Source Bin

Stream Multiplexer

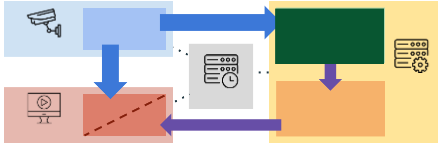
GPU Inference Engine

DeepStream Tracker

DeepStream Analytics

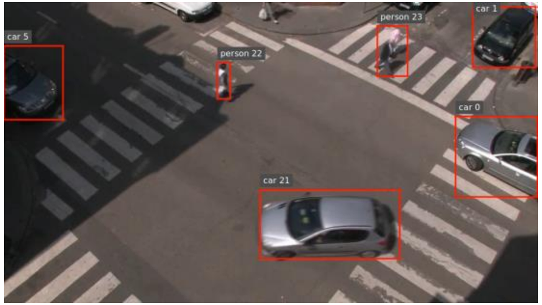
Message Converter

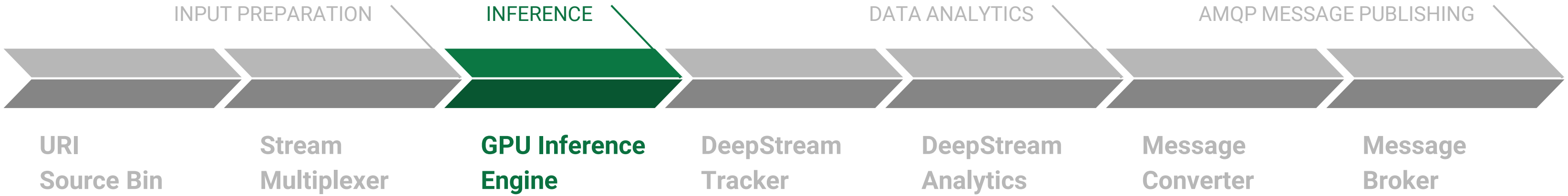
Message Broker



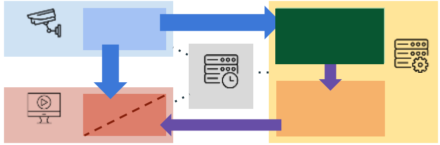
DeepStream Application Pipeline

- Inference is performed using YOLOv3 on frames of video to predict object classes and their locations
- Detected classes, confidences, and bounding box coordinates are saved for each object in each frame



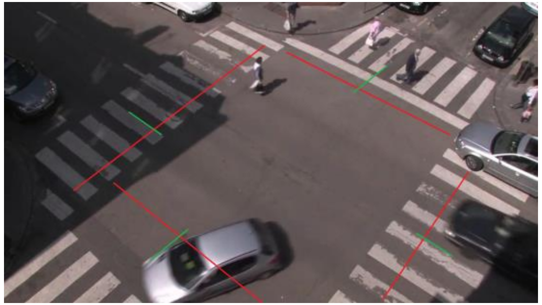


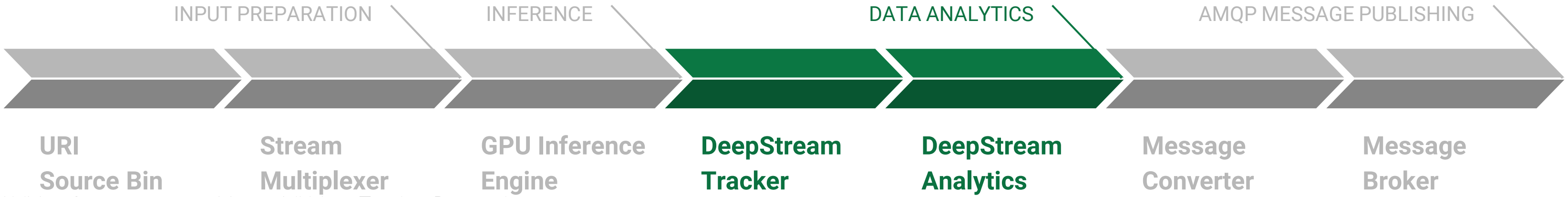
Unedited Video Source: rouen_video.avi (Urban Tracker Dataset)



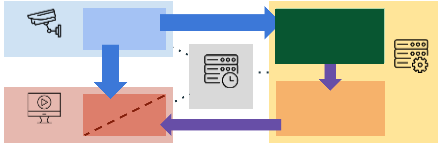
DeepStream Application Pipeline

- Detected objects are tracked and given ID numbers
- Analytics are performed on objects to determine...
  - their current direction of travel
  - whether they've crossed any user-defined boundaries



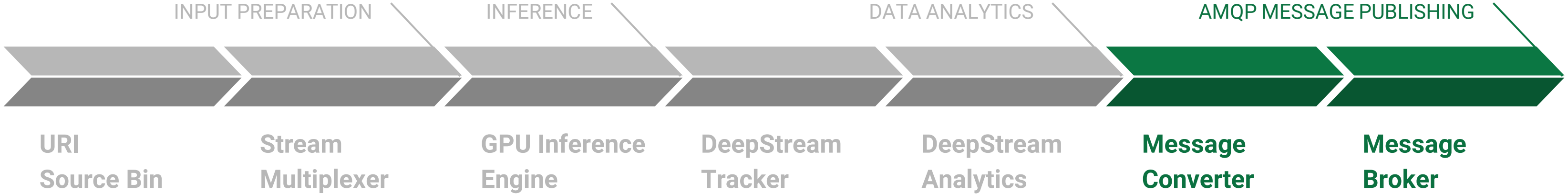


Unedited Video Source: rouen_video.avi (Urban Tracker Dataset)



DeepStream Application Pipeline

- Message payloads are generated from data recorded during previous steps
- Publisher is configured and used to send messages to the AMQP broker



INPUT PREPARATION

INFERENCE

DATA ANALYTICS

AMQP MESSAGE PUBLISHING

URI Source Bin

Stream Multiplexer

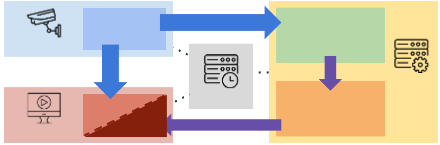
GPU Inference Engine

DeepStream Tracker

DeepStream Analytics

Message Converter

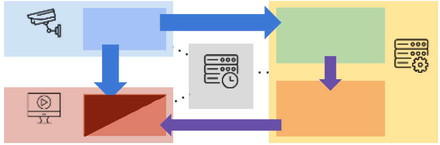
Message Broker



Viewing Client - Writer Thread

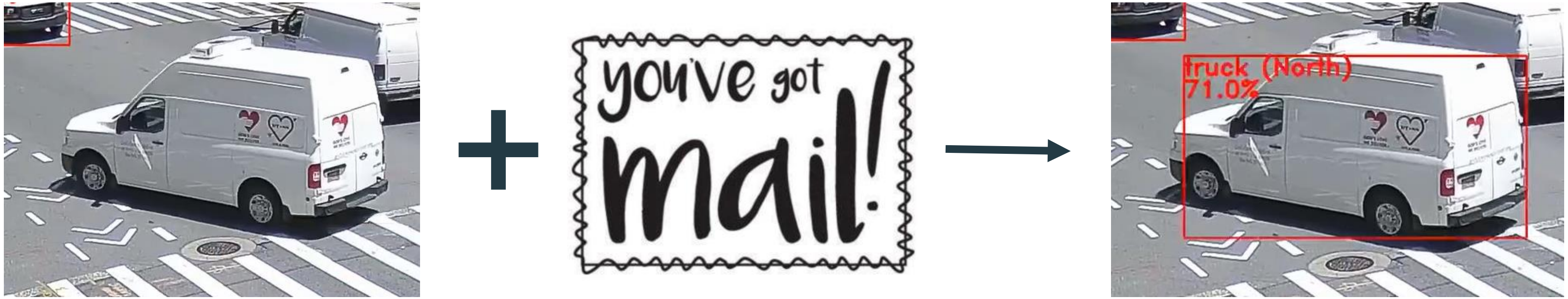
Gets information from message broker and **WRITES** information to reader

1. Receives data from the message broker
2. Picks out important information received in the message
3. Passes on important info to the reader thread



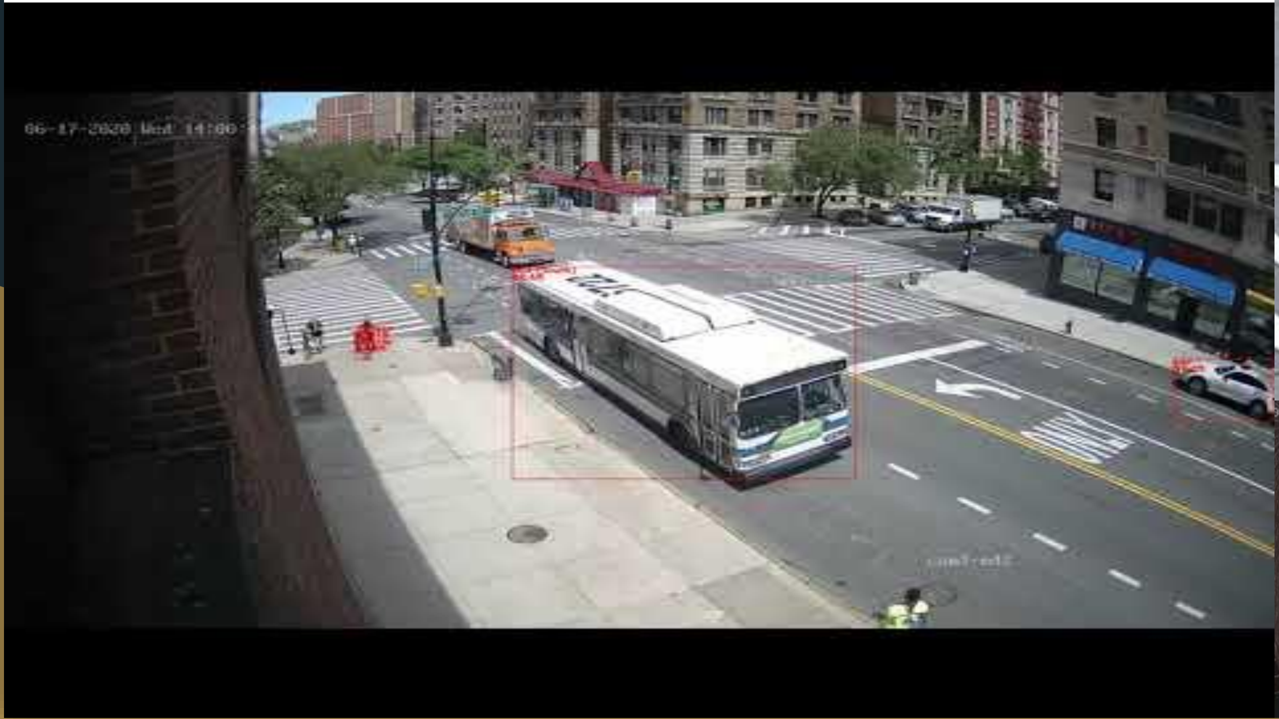
Viewing Client - Reader Thread

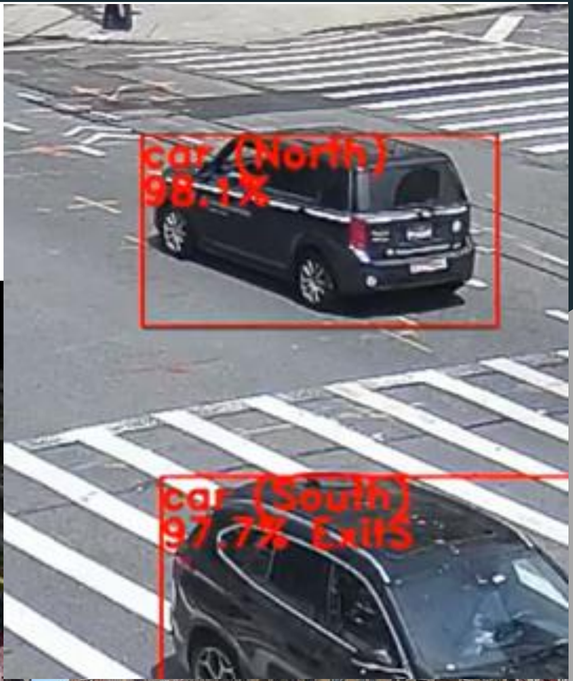
- **READS** information from writer
- Performs specialized task: draws boxes around objects and labels them

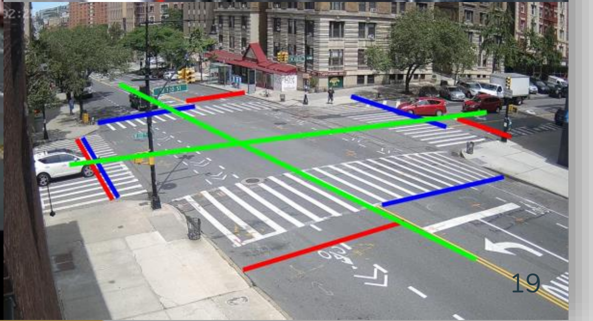


http://www.winlab.rutgers.edu/ https://bzz3ru.wixsite.com/smartintersection

Viewing Client Combined Output







http://www.winlab.rutgers.edu/ https://bzz3ru.wixsite.com/smartintersection

Next Steps

- Train a custom YOLO network designed specifically for vehicle detection
- Upgrade object detector from YOLOv3 to YOLOv4
- Adapt project to work with point cloud data

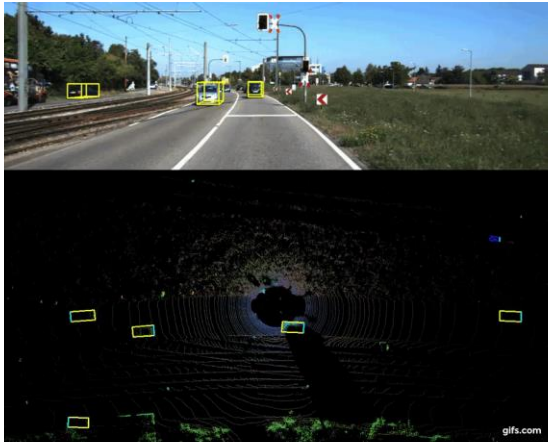


Image Source: https://github.com/ghimiredhikura/Complex-YOLOv3

Questions?

For technical details and more in-depth explanations of our project, check out our website and WINLAB Wiki page!

bzz3ru.wixsite.com/smartintersection

www.orbit-lab.org/wiki/Other/Summer/2020/SmartIntersection

To our advisors, WINLAB staff, and fellow interns…

…on behalf of the entire Smart Intersection team, thank you for a wonderful learning experience and a great time during the Summer 2020 Internship!