Reinforcement Learning-based Traffic Control to Optimize Energy Usage and Throughput (CRADA report)

Thomas P. Karnowski
Matthew Eicholtz
Wael Elwasif
Regina Ferrell
Thomas Naughton
Sean Oesch
Robert Patton
Derek Rose
Ryan Tokola
Steven Young

CRADA final report for
CRADA number NFE-18-07107

Approved for public release.
Distribution is unlimited.

April 2019

OAK RIDGE NATIONAL LABORATORY
MANAGED BY UT-BATTELLE FOR THE US DEPARTMENT OF ENERGY

Electrical and Electronics Systems Research Division

RE REINFORCEMENT LEARNING-BASED TRAFFIC CONTROL TO OPTIMIZE ENERGY USAGE AND THROUGHPUT

Thomas P Karnowski
Matthew Eicholtz
Wael Elwasif
Regina Ferrell
Thomas Naughton
Sean Oesch
Robert Patton
Derek Rose
Ryan Tokola
Steven Young

July 2019

CONTENTS

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The US roadways are critical to meeting the mobility and economic needs of the nation. The United States uses 28% of its energy in moving goods and people, with approximately 60% of that utilized by cars, light trucks, and motorcycles. Thus, improved transportation efficiency is vital to America's economic progress. The increasing congestion and energy resource requirements of transportation systems for metropolitan areas require research in methods to improve and optimize control methods. Coordinating and optimizing traffic in urban areas may introduce hundreds of thousands of vehicles and traffic management systems, which can require high performance computing (HPC) resources to model and manage. In this work, we seek to use machine learning, computer vision, and HPC to improve the energy efficiency aspects of traffic control by leveraging GRIDSMART traffic cameras as sensors for adaptive traffic control, with a sensitivity to the fuel consumption characteristics of the traffic in the camera's visual field. Traffic control use cases using reinforcement learning have been published and achieved good results. Surveys from DOE national laboratories estimate that the fuel cost of idling is 6 billion gallons wasted annually [Argonne National Laboratory, 2019]. GRIDSMART cameras—an existing, fielded commercial product—sense the presence of vehicles at intersections and replace more conventional sensors (such as inductive loops) to issue calls to traffic control. These cameras, which have horizon-to-horizon view, offer the potential for an improved view of the traffic environment which can be used to generate better control algorithms.

1.   STATEMENT OF OBJECTIVES

There are two primary objectives in this project. The first is to develop algorithms that essentially teach GRIDSMART cameras to estimate fuel consumption of vehicles in their visual field. The second is to use this capability to control traffic lights to improve energy efficiency by adaptively changing light timing and phasing, while minimizing penalties to throughput and mobility. High performance computing (HPC) can play a role in both objectives by allowing more complete exploration of the machine learning architectures, parameters, and methods that enable the ability to determine vehicle types. HPC-based simulations that model traffic and capture the performance of GRIDSMART cameras in estimating the visual field (extrapolated from real data using developed algorithms and models) serve as training and testing data for reinforcement learning algorithms that learn policies for traffic camera control. The planned key outcome of this work will be control strategies generated through a novel reinforcement learning framework, with performance measured through simulations and validation data and oriented toward the GRIDSMART sensing capability. Other important outcomes include projections of the required sensing capabilities to achieve these control strategies. This will pave the way for future research to expand the number of studied intersections, investigate the potential of wide-range coordinated control, add naturalistic driving study data for higher resolution and simulation detail, extend sensing capabilities to other technologies such as radio-frequency identification (RFID)/cellular and/or connected vehicle technology, and incorporate direct vehicle emissions sensing to minimize cumulative emissions measured.

2.   BENEFITS TO FUNDING DOE EERE'S MISSION

The Department of Energy Office of Energy Efficiency and Renewable Energy (DOE EERE) has a stated mission: "The mission of EERE is to create and sustain American leadership in the transition to a global clean energy economy. Its vision is a strong and prosperous America powered by clean, affordable, and secure energy" [EERE, 2019a]. The Vehicle Technologies Office (VTO) within EERE features multiple program areas, including the Energy Efficient Mobility Systems (EEMS) Program, which "envisions an affordable, efficient, safe, and accessible future in which mobility is decoupled from energy consumption. The program will conduct early-stage R&D at the vehicle, traveler, and system levels, creating new

knowledge, insights, tools, and technology solutions that increase mobility energy productivity for individuals and businesses" [EERE, 2019b]. The project objectives focus on adapting existing technology to improving intelligent transportation infrastructure operations and developing tools to improve traffic in urban areas. Finally, as an HPC4Mobility seedling project, the project sought to bring HPC resources to bear and create technology solutions that are beyond human observation capacity. These objectives improve the energy efficiency of the American transportation system and thus meet the goals of both EERE and VTO. Furthermore, the problems posed here are worthy of a national laboratory, where proper resources can be pooled and applied to these challenges for the betterment of the United States and possibly internationally.

3. TECHNICAL DISCUSSION OF WORK PERFORMED BY ALL PARTIES

There were two main technical focus areas in this project, broadly described as data collection and simulation. These are aligned with our two aspirational goals of "teaching the GRIDSMART camera to estimate fuel consumption" and "teaching a grid of cameras to improve traffic mobility and fuel economy".

3.1 DATA COLLECTION TECHNICAL FOCUS

Teaching a camera to estimate fuel consumption is an aspirational goal, but it has two fairly straightforward components. The first is to attempt to recognize the vehicle type, as different vehicles have different fuel consumption characteristics. The second is concerned with vehicle dynamics and driving styles.

Some observations can be easily made that the vehicle dynamics are more important and fuel consumption models use both vehicle characteristics and dynamic information [Rakhra, 2011; Berry, 2010]. We elected to pursue the vehicle visual characteristics as our primary technique for two main reasons: first, vehicle dynamics would largely be a function of vehicle tracking with requirements for calibrated cameras, which are not readily available based on conversation with GRIDSMART, and second, progress toward the problem of vehicle visual classification has desirable side benefits, such as providing insight into traffic constituents and characteristics.

Machine learning for visual object recognition has made great advances in recent years with the publishing of large labeled data sets and deep algorithms that leverage both GPU acceleration and large data sets [Goodfellow et al., 2016]. The technology has become ubiquitous, and consequently there are many potential vendors of solutions for image-based sensing. However, a large problem is the lack of data to customize and train algorithms to attain the accuracies desired. Therefore, with GRIDSMART advice, we elected to start our effort with a data collection where we could identify vehicle types with a relatively high-precision, high-resolution image and map these classifications onto the GRIDSMART overhead camera view.

The data collection technical focus consists of collection methodology and locations; information about the ground-based sensor (GBS); the "ground truth" vehicle classification; vehicle segmentation; the data set we created and the "data hygiene" necessary; and some analysis using the data set. We also discuss an alternate data set we used as a contingency and variations encountered that we believe help define limitations on visual-based vehicle fuel estimates. Finally, we briefly describe data from a naturalistic driving study that segue ways into our simulation discussion.

3.1.1 Collection methodology and locations

ORNL has been developing a ground-based sensor (GBS) system that could be used as a system for the high-resolution image collection. The fundamental concept is illustrated in Figure 1, which has been presented in past VTO Annual Merit Review (AMR) events. The GBS captures a relatively high resolution image of the vehicle of interest near a GRIDSMART instrumented intersection. Simultaneously, the same vehicle is captured from the GRIDSMART overhead view. The GBS image is classified by some method (manual, the use of auxiliary information such as license plates, or a machine-learning based object classification), which creates labels for the overhead GRIDSMART-captured images.
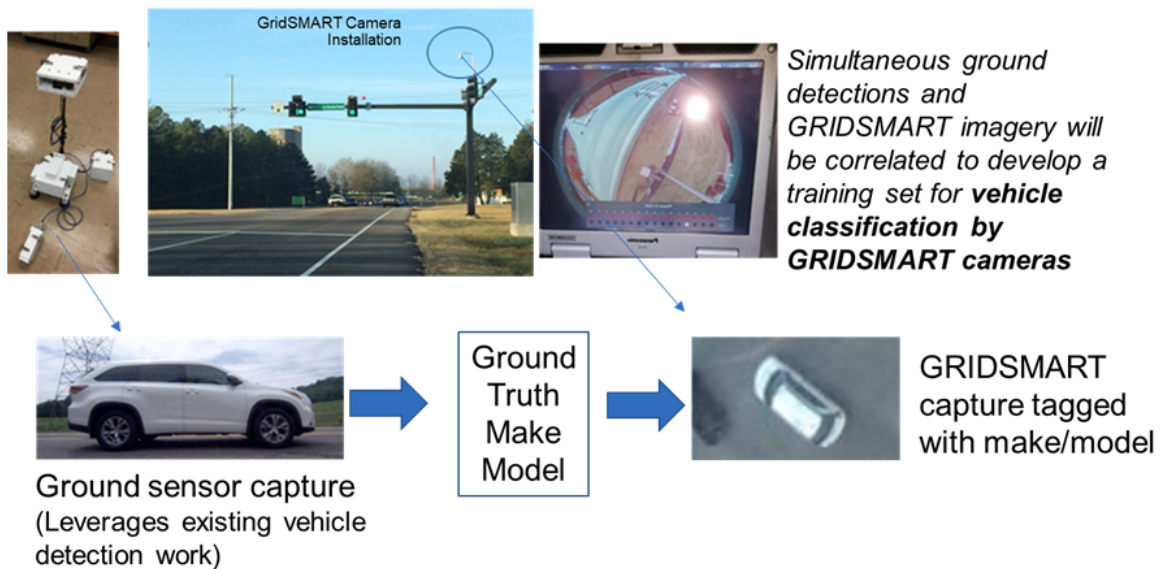
Figure 1. Collection concept to build ground-truth data. A ground-based sensor (GBS) is shown in the upper left in a laboratory setting. The sensor is deployed in the field under an intersection with a GRIDSMART camera (top center images); data can be collected using a USB drive connected to the GRIDSMART unit with its own custom interface (top right). On the bottom we see the images collected by the GSB (left) and GRIDSMART (right). Some mechanism is used to estimate vehicle ground truth (make and model), such as manual predictions from the ground sensor image or a "fine-grained" vehicle classification system.

There are three collection locations at ORNL which utilize GRIDSMART cameras, all at intersections on Bethel Valley Road. The first location is at the Spallation Neutron Source (SNS), the second at Melton Valley Drive, and the third at First Street. Sample image data are shown from all three sites in Figure 2. We note that even given that all intersections have their own unique characteristics, the ORNL installations are a little unusual; all are at 3-way intersections, the SNS intersection is largely a side view, and the First Street intersection has a portion of the camera view blocked by an extended pole over the street. Consequently, although we collected some data from all three sites, we focused our efforts on the Melton Valley location.

Data collection consisted of accessing the GRIDSMART unit within the traffic control cabinet and plugging a specified USB drive into the unit. The GRIDSMART equipment would indicate the status, and data would begin collecting on the drive. A 1 TB drive could store roughly 10 days of data, organized by the hour, with roughly 26,000 images per hour (for a collection rate of approximately 7 frames per second). We did not directly address the GRIDSMART camera at the direction of GRIDSMART because they recommended the USB drive method to avoid interfering with the camera function. (In addition,

technical personnel at GRIDSMART indicated the camera was not addressable in any event.) We attempted to use auxiliary tools such as a programmable USB switch by ACRONAME, which we tested at GRIDSMART using a laboratory camera and controller, but this method was not reliable beyond a few days, and we also had concerns with its exposure to the environmental conditions at the traffic cabinet.
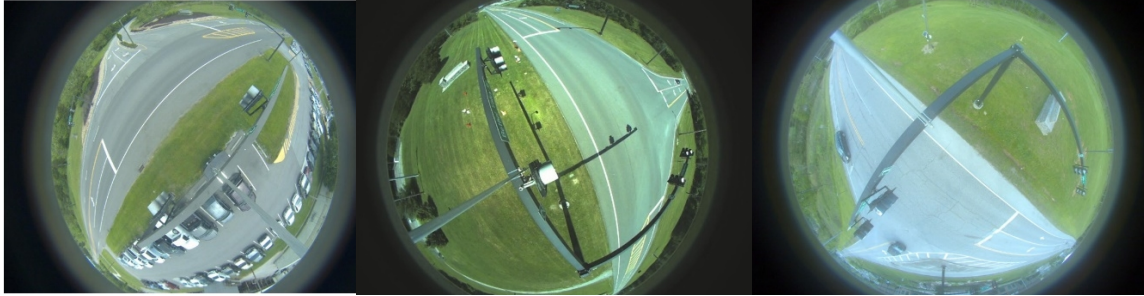
Figure 2. GRIDSMART installations at ORNL. Left: Spallation Neutron Source. Middle: Melton Valley Road. Right: First Street.

3.1.2   Ground-Based Sensor

The ORNL GBS captures multiple images of a target vehicle and uses embedded algorithms to create two primary images, either with native size ("crop") or projected to a common size ("projected"). In our collections, we used both images, but for our data set we limited the collection to the projected images. The images are created so that the wheels are aligned from image to image. The GBS functioned at a fairly high technology readiness level (TRL) when this project was started but still underwent considerable engineering development for environmental hardening, and therefore data collection was sporadic until September 2018. At that point, however, the system operated 24/7 and collected a large amount of data until field operations ceased in November 2018. An automated license plate reader was added to the system after this point but was not utilized for this project.

3.1.3   Ground-Truth Vehicle Classification

With the ubiquity of deep learning applications, especially in visual object recognition, our team was able to identify a commercial application as a candidate for "fine-grained classification" of vehicle make and model. ORNL procured a license for this technology for the duration of the project and used it extensively to deliver a ground-truth label for the GBS images. We note that we submitted both the crop and projected image to the tool, since our experiments had shown that sometimes one classification would work and other times the alternate would deliver a classification. When both presented a classification, they agreed the vast majority of times. We note that there are other fine-grained systems in existence and that methods for fine-grained automobile classification are available in the open literature [Yang et al., 2015; Dehghan, 2017; Sighthound, 2019; Krause, 2013, 2015]. However, the data sets for these sources are mostly sourced from web-based URLs and therefore are typically very different from the automated "real world" images we collected from the GBS, let alone the GRIDSMART overhead image view.

3.1.4   Vehicle Segmentation and Tracking

Segmentation in image processing refers to algorithms that separate an object of interest (such as a vehicle) from the background that is not of interest. There are modern, state-of-the-art deep learning approaches called "semantic segmentation" which are becoming increasingly powerful [e.g., Badrinarayanan (2017)], but in our work we sought to emulate, to some degree, the GRIDSMART image processing pipeline by using computer-vision-based tools that perform vehicle segmentation in real time. We note that the GRIDSMART segmentation was not available to our work, but we were able to create a

reasonable approximation to their process that allowed us to model their camera performance. We used the MATLAB computer vision toolkit [MathWorks, 2019a], although open-source toolboxes such as OpenCV [OpenCV, 2019] could have been a viable alternate.

Our initial work, prior to obtaining many GBS images, focused on a pipeline for reducing the GRIDSMART data from a field collect into a set of segmentation images. The pipeline operated on data 1 hour at a time since the GRIDSMART data was organized into hourly folders. (Note that it was not our goal to segment the entire data stream to detect vehicles at the precision that GRIDSMART requires for traffic sensing; rather, we simply were trying to formulate a method to build a data set of ground truth and overhead imagery.) The initial work generated a background model from the hour of data and then performed a coarse vehicle segmentation on every Nth frame (with N set experimentally between 5 and 10), with a goal of finding frames with a high probability of vehicles. These frame with a high probability of detection were then folded into a MPEG file to facilitate data processing and compressing data, as accessing the individual frames of data stored on a server had a lot of network file access overhead. The next step was a more refined vehicle segmentation, followed by a tracking process based on Kalman filtering and yielding vehicle detections. An example of an image sequence of a tracked vehicle is shown in Figure 3.

Figure 3. Example of vehicle tracked with Kalman filter from the GRIDSMART camera point of view. The image degradation due to resolution and the fisheye lens is profound at longer distances.

The segmentation process used a mixture of Gaussian model to separate potential vehicles from the background. The implementation used the MATLAB computer vision ForegroundDetector [MathWorks, 2019b] with parameters set to 25 learning frames, a learning rate of 0.005, a minimum background ratio of 0.7, automatic initial variance, and a Gaussian cardinality of 3. Foreground detection was followed by a

sequence of screening and post-processing, including an image dilation and an estimate of the vehicle location based on an observed trajectory map for the location. The trajectory map was created for the Melton Valley intersection and included vehicles proceeding left-right on the image, vehicles proceeding right-left, and vehicles entering left and turning to the right.

While we believed the segmentation process was useful, when we began to collect GBS data we elected to use the GBS collections as a screening process and only perform vehicle segmentation on the GRIDSMART imagery to target a particular GBS image. Each GBS image was tagged with the estimated vehicle class. We then identified a range of likely GRIDSMART frames that would contain the GBS collect and pulled them from the GRIDSMART data. This allowed us to focus on the data set creation more effectively. We still used the same parameters and methods for the vehicle segmentation, but we had to refine the process by manually selecting the candidate vehicle in the GRIDSMART view due to timing offsets. The synchronization between the GRIDMART imagery and the GBS system was initially believed to be straightforward, but later we found that this was not the case. Initially, neither GBS nor the GRIDSMART cameras at ORNL were time-synced to GPS time. The GBS system was corrected early in the project, but we found that the GRIDSMART cameras were not synced, and therefore (with the assistance of GRIDSMART) we were able to provide a time-base for the cameras. Even at that point, however, we found disagreement and drift between the two systems that required tools for manual synchronization. Our goal was to identify the GRIDSMART image that corresponded to the GBS image. In many cases, once one pair of images was synchronized, the next pair of images was essentially identical in time offset for a long period of time but periodically needed to be readjusted (presumably due to clock drift.) In many cases it was easy to make this adjustment, but on occasion vehicles with very similar appearance made it more challenging. Cases with unknown collections were simply omitted (both initially and through a later quality check).

Additional data hygiene was required beyond this step, including the selection of the vehicle of interest, an initial selection of the vehicle direction (due to modifications to the software, early captures from the GBS did not always supply a vehicle direction), and lane. As we accumulated examples we were able to perform accurate estimates of the direction and lane using the coordinates chosen and a nearest-neighbor classifier; a deep convolutional neural network based on Alexnet [Krizhevsky et al., 2012] that estimated the direction of travel; and an estimate from the ground image itself, which showed some variation based on the direction and lane. In the end, a set of GBS-GRIDSMART "fusion" events were created that served as correlated material for our data set.

A second process was then created to attempt to identify the highest resolution frame possible for each collection fusion. In this process, the vehicle of interest was selected using the ground-truth process where a point was placed on the vehicle segmentation "blob". The vehicle was then tracked using speeded-up robust features (SURF) [Bay et al., 2006] implemented again in the MATLAB computer vision toolbox. In this algorithm, points that are likely to be highly unique are identified in subsequent frames, and then the points are matched to perform a tracking function. When the vehicle was closest to a selected point based on the lane of travel, the process was completed and the best frame was saved along with information such as the vehicle bounding box, oriented length, and collection time. This sequence is depicted in Figure 4 and 5.

Figure 4. Example of vehicle segmentation with selected bounding box. In this example, there is only one object in the field of interest, but generally this was not the case, so the object of interest matching the GBS (inserted in lower left) was chosen manually.

Figure 5. Left: vehicle with SURF points identified. Right: result of tracking through matching SURF points.

In addition to GBS-based collections, there were a number of unique vehicle types that passed through the intersection that we also identified through a segmentation process. These included larger vehicles that were not identified in the GBS sensor; we relied on our original segmentation process to identify these, given that they could be screened initially by the size of their bounding box and including "18-wheelers", large multi-axle trucks, motorcycles and bicycles, and busses. We termed this segmentation process the "Wild non-GBS" data (WGBS). We were also able to identify vehicles that were used routinely at ORNL, including passenger utility vehicles such as the Chevrolet Express minibus and delivery vans. These vehicles typically lacked a year/make/model but were included for their utility.

3.1.5 Data Set, Characteristics, and Utility Examples

In the resultant data set, a total of 6,695 vehicles were identified in 685 different vehicle categories. The vehicle distribution is not evenly spread, as roughly 150 classes have a single vehicle and some have as many as 238. The median number in each category is only 4, which limits the utility of the data set for methods such as deep learning, but in this section, we show there are multiple ways to utilize data of this kind.

The WBGS data includes the special vehicle classes of 18Wheeler, Bus, Chevrolet Express Bus, DeliveryVan, Ford Transit XLT, FordSuperDuty, Cycle, and MultiAxle. These seven classes were isolated due to their size, estimated MPG, or common occurrences through the ORNL intersections (e.g., the Chevrolet Express Bus is used by the ORNL fleet to assist employees and visitors to the ORNL campus). A distribution of the 20 largest classes is shown in Figure 6. Note that we make no claims about how this distribution truly reflects the "vehicle demographics" at ORNL, as we focused on detections from the GBS and our special classes.

Another note regarding the data is the uniqueness of each vehicle. We also make no claims that each image is truly an independent vehicle sample; in other words, while the image may be taken at a different day or time, two images of a 2005–2011 Toyota Tacoma may indeed be the SAME VEHICLE. We still believe that variation in the environmental conditions and possibly vehicle location make this potential duplication worthwhile, but the only true way to avoid such issues would be to confirm single-vehicle entries using technology like automated license plate readers (ALPRs) in the collection process. We did attempt to prevent images that were too close in time from appearing in the data set (e.g., two images separated by a few seconds or less were deleted). Finally, we also reviewed each image and each GBS-GRIDSMART pair to ensure the classifications seemed correct and that the GBS-GRIDSMART images were the same vehicle. However, we must allow for the possible errors in the data collection and screening process, and we are hopeful that if the set is released, any errors found by researchers will be shared for future corrections.
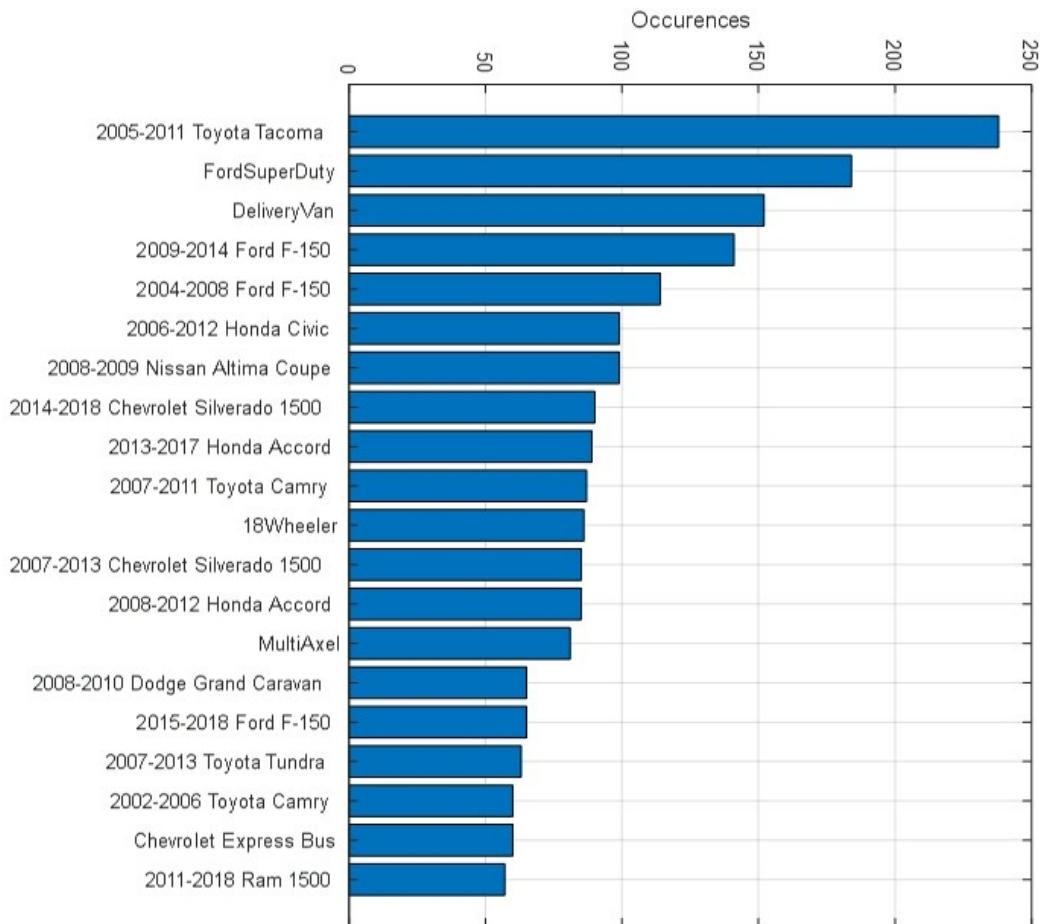
Figure 6. Vehicle classes ranked by number of entries (showing the largest 20).

While our raw data includes the entire GRIDSMART image, for distribution we focus on an intersection view as well as the vehicle segmentation mask. The mask is an image that is the same size as the intersection view but is binary valued, with white indicating pixels where we expect a vehicle and black are background pixels. Some example images are shown in Figure 7 through Figure 11, along with the segmentation mask and a fusion image that highlights the segmentation process.
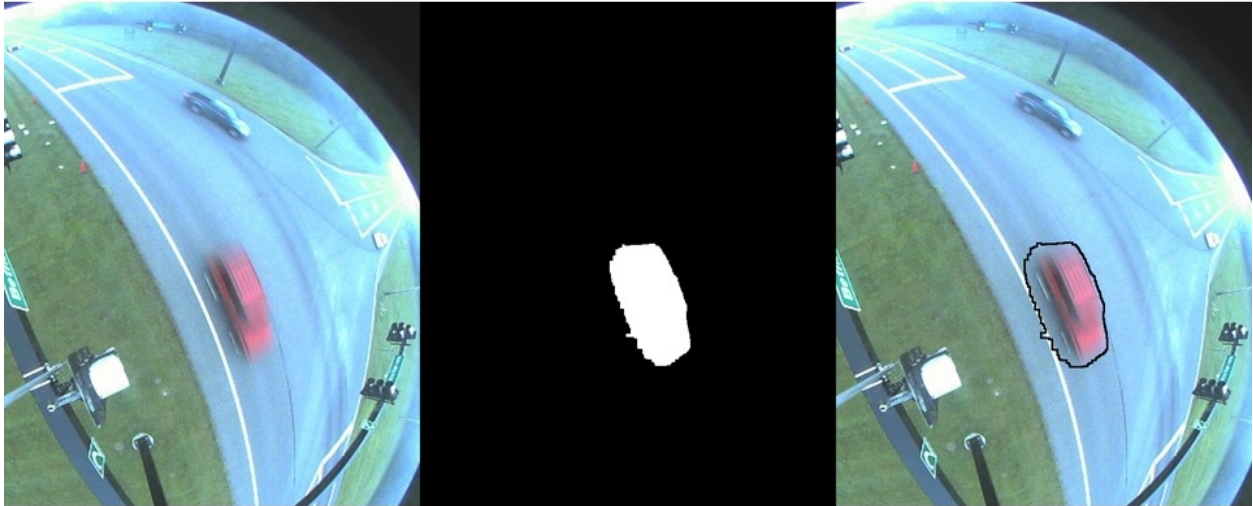
Figure 7. Example of intersection image (left), segmentation mask (middle), and superimposed segmentation mask for 2001 Jeep Wrangler. The image was taken under lower light conditions and has some image blur as a result. Little shadow is present, however.
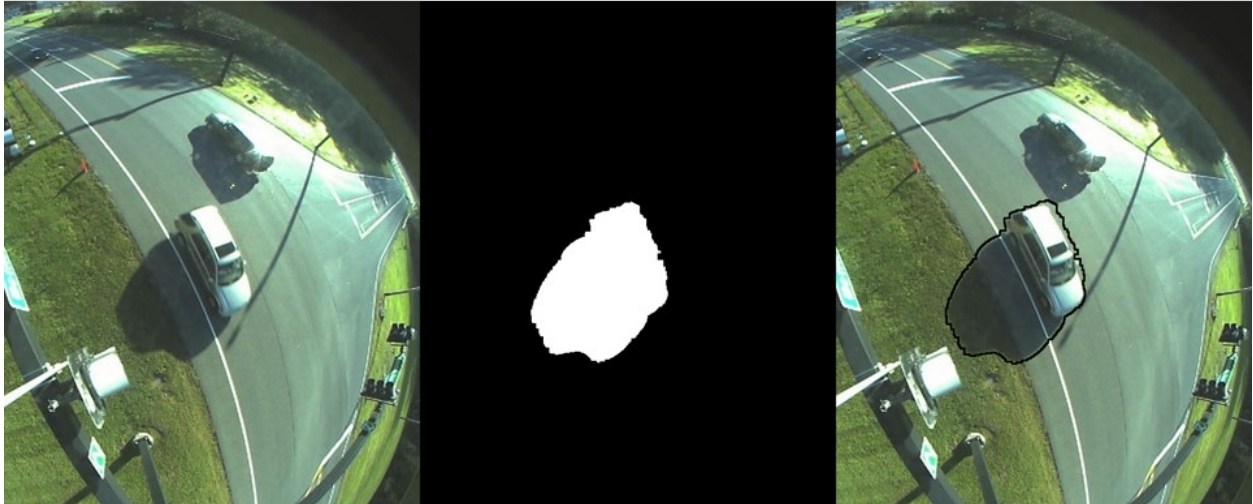
Figure 8. Images of Mercedes-Benz M-class vehicle from intersection view (left), segmentation mask (middle), and superimposed image (right). In this case there is excess shadow from the early morning sunrise, causing an issue with the mask; the oriented bounding box reveals a wholly inaccurate orientation of approximately 60 degrees, which would normally be indicative of a vehicle traveling from the bottom left of the image to the top right, or vice versa.
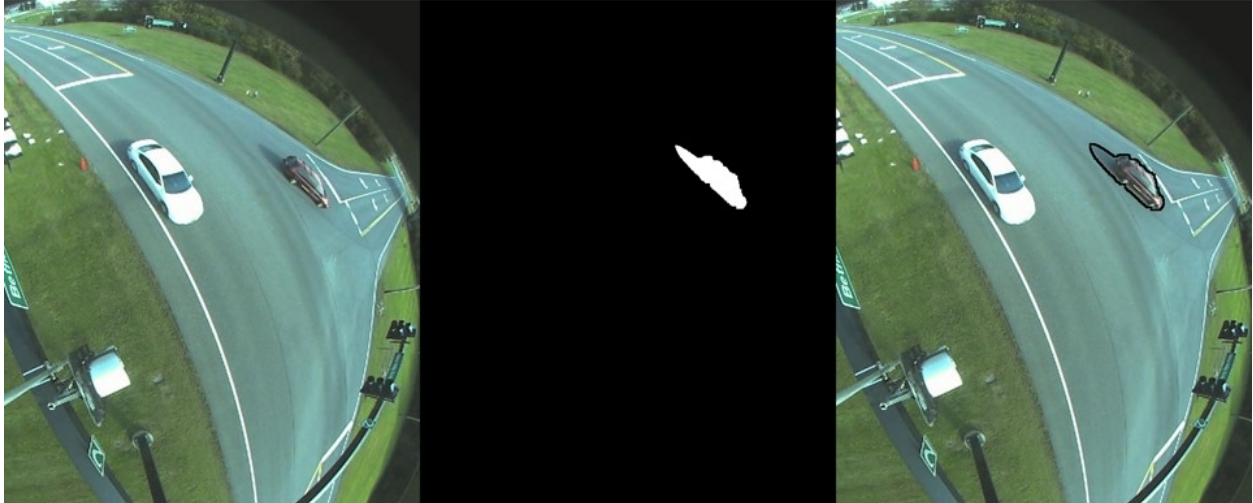
Figure 9. Example of vehicle from GRIDSMART view: left, the intersection image; middle, the segmentation mask; and right, the segmentation superimposed on the vehicle. In this case the orientation (roughly -40 degrees) is aligned with the vehicle for the most part, despite the "tail" from the shadow. The vehicle was classified as a 2001–2007 Dodge Caravan.
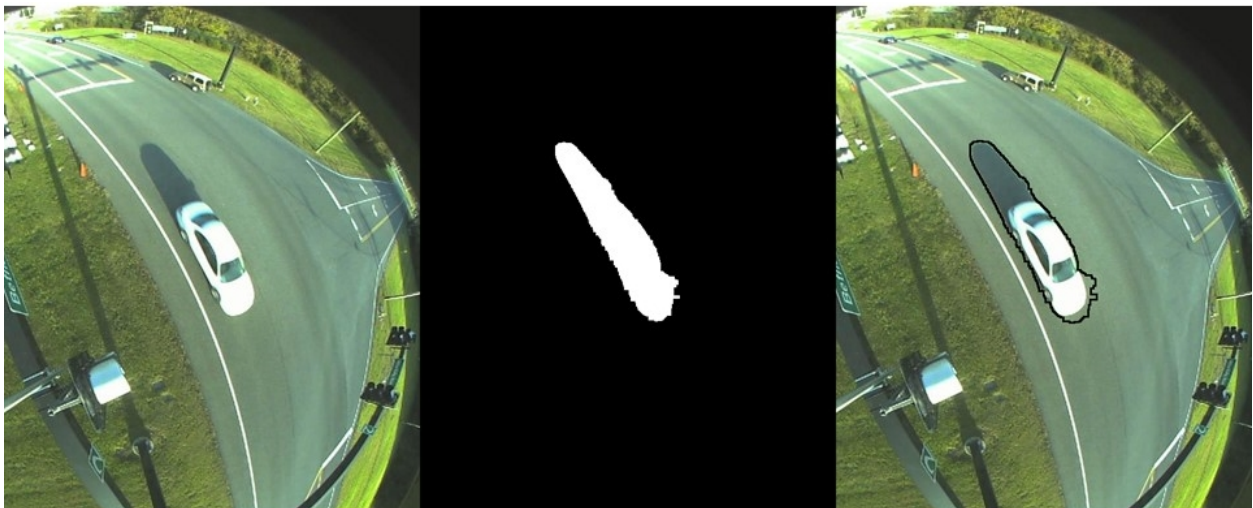
Figure 10. Another example of a collect with a significant shadow component. In this case the shadow does not impact the bounding box orientation but has a large impact on the effective size estimate, nearly doubling the estimate. This vehicle is a 2004–2007 Ford Taurus. Left: intersection view; middle: segmentation mask; right: intersection view with superimposed segmentation mask.
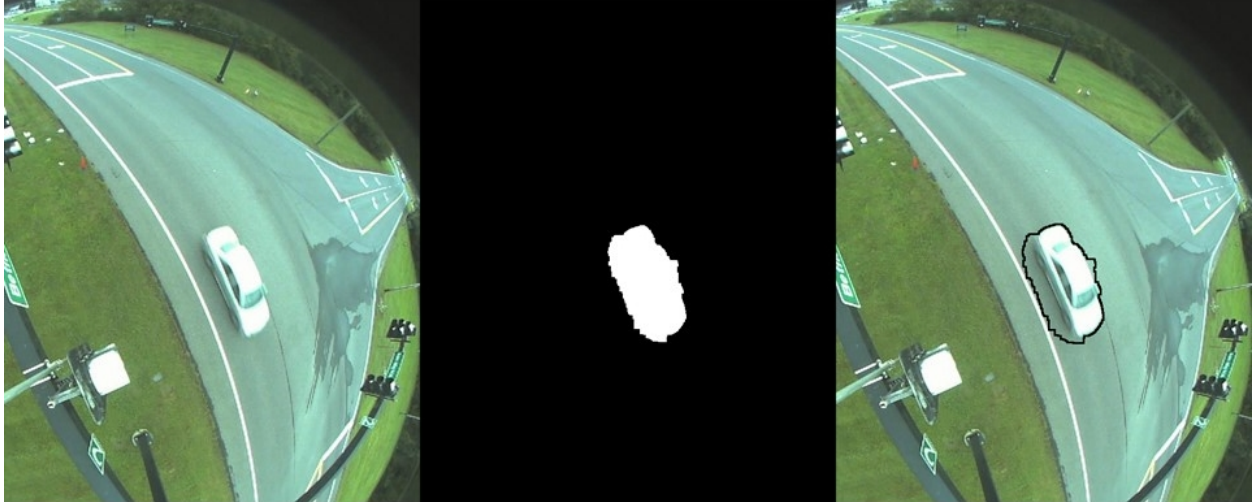
Figure 11. Collection under cloudy conditions. A low spot with standing water and damp pavement is seen in the mid-right-center region of the image. This vehicle is a 2009–2013 Toyota Corolla. Left: intersection view; middle: segmentation mask; right: intersection view with superimposed segmentation mask.

A table of meta-data associated with the data collections is also included in the data set. The headings are described in Table 1 for the full data set; however, the full release will feature a subset of this data to simplify presentation.

Table 1. Full data set headings.

Variable Name	Description
ClassSource	Identifies the classification origin for the class column. There are three possible values: "Wild", which refers to the WGBS classes, "NAPass", which refers to vehicle images where additional methods were taken to augment the automated identification, or "ClassPass", which utilized the automated identification method. Note that due to a variety of steps taken in the classification process, the proportion of NAPass and ClassPass cannot be interpreted as an indication of the performance accuracy of the automated identification methods.
Class	Vehicle class, either from automated identification methods with human verification or from the WGBS entries.
IntersectionFile	The name of the image for the intersection view of the vehicle of interest.
IntersectionFileSeg	The name of the segmentation file for the intersection view of the vehicle of interest.
GroundFile	The name of the ground image captured by the GBS for the vehicle of interest
CroppedFile	The name of the "cropped" image, which was obtained by using the bounding box columns from the intersection image. The cropped image was then resized to 127 pixels width and 127 pixels height.
BBoxX	The bounding box "X" corner coordinate in the intersection view image.
BBoxY	The bounding box "Y" corner coordinate in the intersection view image.
BBoxW	The bounding box width in the intersection view image.
BBoxL	The bounding box height in the intersection view image.

Table 1. Full data set headings (continued).

Variable Name	Description
OrientationInDegrees	The orientation of the segmentation image in degrees, where 0 degrees is oriented horizontal (i.e., traverses the image field left to right or right to left) and 90 degrees is oriented vertical (i.e., traverses the image field top to bottom or bottom to top).
OrientedLength	The length of the vehicle segmentation mask bounding box after correcting for the orientation.
OrientedWidth	The width of the vehicle segmentation mask bounding box after correcting for the orientation.
CentroidX	The X coordinate of the centroid of the vehicle bounding box.
CentroidY	The Y coordinate of the centroid of the vehicle bounding box.
UTCCollectionTime	The collection time of the ground-based image in UTC. When a ground-based image was not available, the time was based on the GRIDSMART time. (These times should be accurate to within +/- 2 seconds or less; the intention is to capture the rough time of day.)
Index	An index number of the entire image for debugging purposes
ClassSourceNumber	A value 1, 2, or 3 for the ClassSource
IndexInClassSource	Index of the vehicle image within the larger ClassSource or ClassSourceNumber set, for debugging purposes.
AllGround	The name of the ground image in a folder that contains all ground images for debugging purposes.
AllCrop	The name of the intersection cropped image in a folder that contains all intersection cropped images for debugging purposes.
AllIntersection	The name of the intersection image in a folder that contains all intersection images for debugging purposes.
NumberOfEntriesInFuelEconomy	Numeric value that indicates the number of vehicle variants that likely fall under this vehicle class (for the non-WGBS classifications). For example, a 4-wheel drive and 2-wheel drive version of the same vehicle would count as two variants.
MeanCombMPG	The mean combined MPG value from the fueleconomy.gov [EERE, 2019c] or alternate fuels data center download [EERE, 2019d] across all the variants for a vehicle type
STDCombMPG	The mean combined MPG value from the fueleconomy.gov download across all the variants for a vehicle type
MaxCombMPG	The maximum value of combined MPG value from the fueleconomy.gov download across all the variants for a vehicle type
MinCombMPG	The minimum combined MPG value from the fueleconomy.gov download across all the variants for a vehicle type
MedCombMPG	The median combined MPG value from the fueleconomy.gov download across all the variants for a vehicle type
NumberOfEntriesWithConventionalFuel	The number of entries using conventional fuel in the fueleconomy.gov download across all the variants for a vehicle type. This number will be less than or equal to the NumberOfEntriesInFuelEconomy column value.

Some general comments regarding vehicle classification from different points of view are in order at this point. An examination of the side view of different vehicle types reveals a variety of features that could potentially discriminate vehicles. This includes the overall body shape, but there are also a variety of features that could be utilized, such as the window shapes, different styling on the doors, the shape of the wheel wells, and other features on the vehicle itself such as body indentations which are largely cosmetic but consistent from manufacturers. Many of these features are not available from the GRIDSMART view, and therefore our estimates on vehicle discrimination from GRIDSMART imagery are highly speculative without more imagery. Our most firm findings are based on the vehicle bounding box and the ability to discriminate vehicle type within those constraints.

Another interesting issue is the degree of variance in pose/position that is available from the GRIDSMART camera. Figure 3 shows an example of a vehicle from different positions, and while the main intent of that image sequence is to represent the variation in resolution, we also note that the image changes from a largely frontal view, to an overhead view, to a passing view that is overhead and slightly behind the vehicle.

The data set developed has multiple applications, including for segmentation studies, shadow analysis, and other computer vision applications. We also show the utility of the set for estimating fuel and vehicle characteristics in the following examples, based on the bounding box size and then by developing a classifier based on the bounding box and the visual characteristics of the vehicle within the bounding box.

While it is possible to calibrate a GRIDSMART camera, in practice this is not always needed, and in this example we simply leveraged the common locations of the intersection images. A heat map of the intersection data is shown in Figure 12. There are three main locations where the images were taken, which we refer to as "NearLane" (A), "TurnLane" (B), and "FarLane" (C). We took the region around these core areas and analyzed the vehicles within each region. A threshold on the oriented bounding box length was set, and the average fuel economy of all vehicles above the threshold was computed. These plots are shown in Figure 13 through Figure 15. These plots show that we can functionally discriminate between high fuel consumers and lower consumers (i.e., "average" vehicles) simply on the basis of the oriented bounding box length. This finding shows that there is value in the visual estimates made virtually "for free" from vehicle segmentation such as that which GRIDSMART already produces.
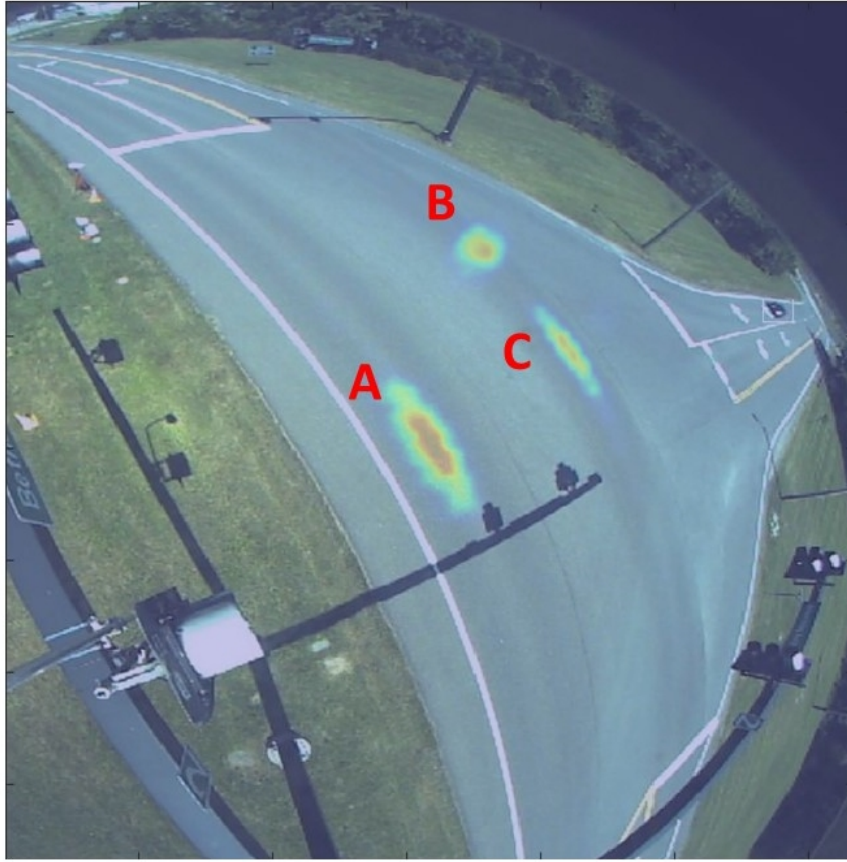
Figure 12. Heat map of vehicle positions from the data set. "A" corresponds to vehicles traveling upper left to lower right; "B" corresponds to vehicles turning toward the right side of the image; and "C" corresponds to vehicles traveling lower right to upper left.
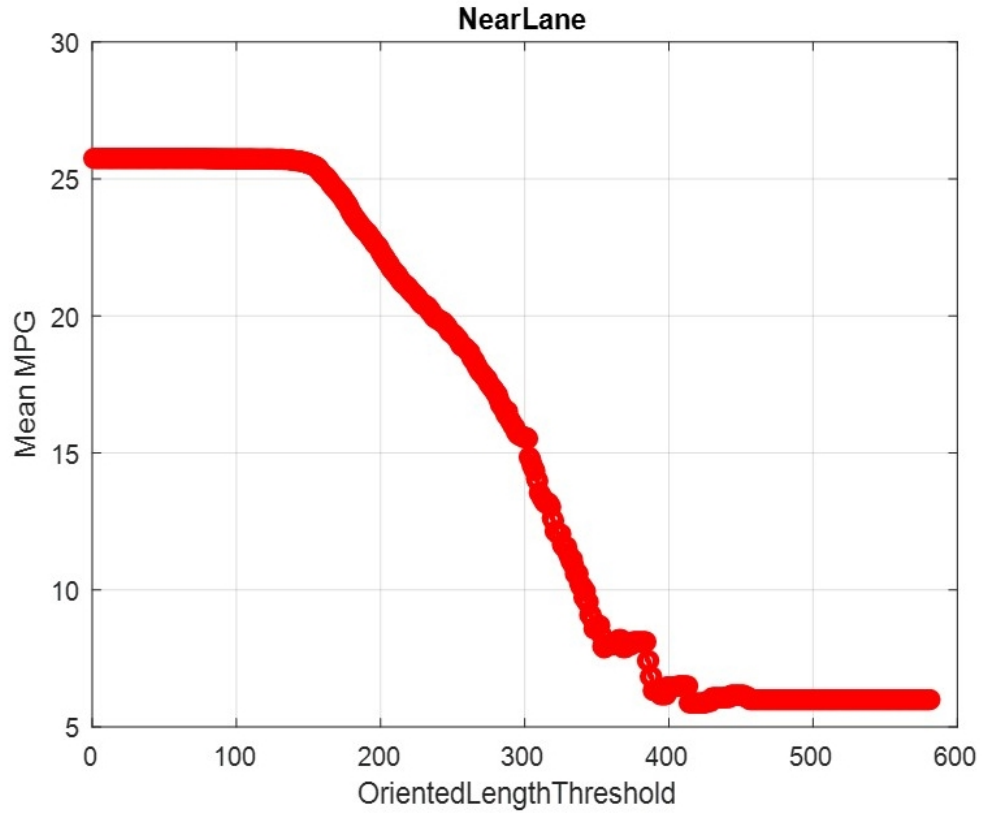
Figure 13. Average fuel consumption estimate in MPG for vehicles in the "near lane" based on the oriented bounding box length. From left to right we see that the average fuel consumption of vehicles with an oriented bounding box length greater than 0 has an average MPG estimate of approximately 26 MPG. The size threshold increases, and we begin to omit smaller vehicles which have better fuel economy in general. Vehicles above ~450 pixels have a mean MPG of approximately 6 MPG.
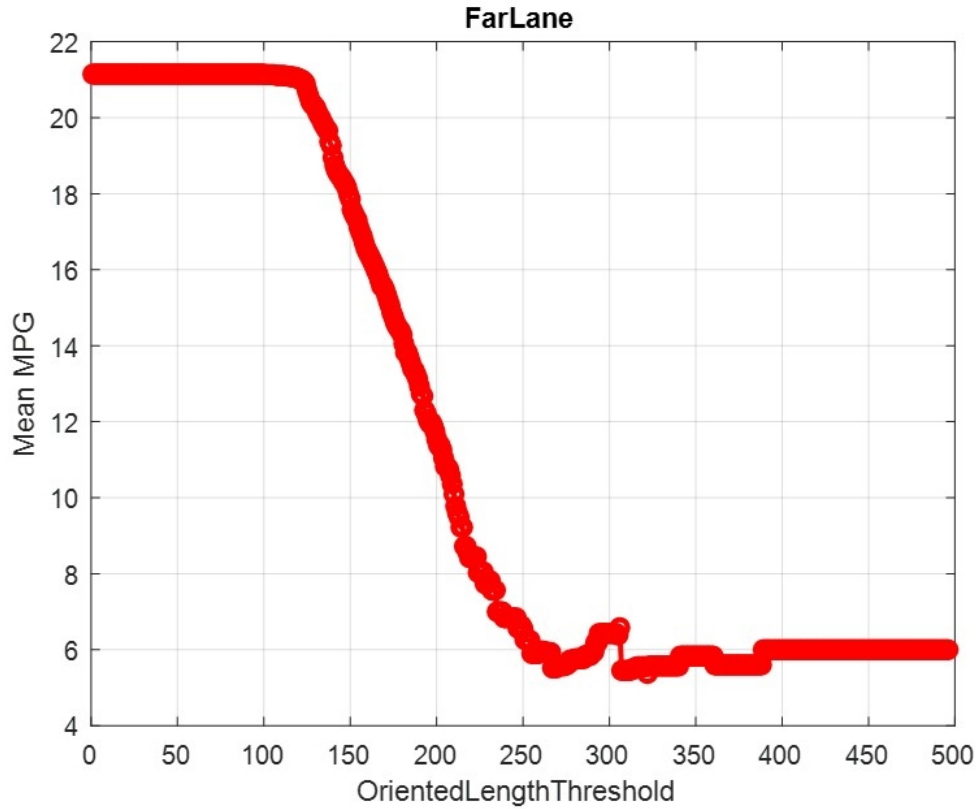
Figure 14. Average fuel consumption estimate in MPG for vehicles in the "far lane" based on the oriented bounding box length. From left to right we see that the average fuel consumption of vehicles with an oriented bounding box length greater than 0 has an average MPG estimate of approximately 21 MPG. The size threshold increases, and we begin to omit smaller vehicles which have better fuel economy in general. Vehicles above ~375 pixels have a mean MPG of approximately 6 MPG. The best explanation for the difference in the mean MPG at lower oriented lengths may be simply differences in vehicle "demographics" for different lanes.
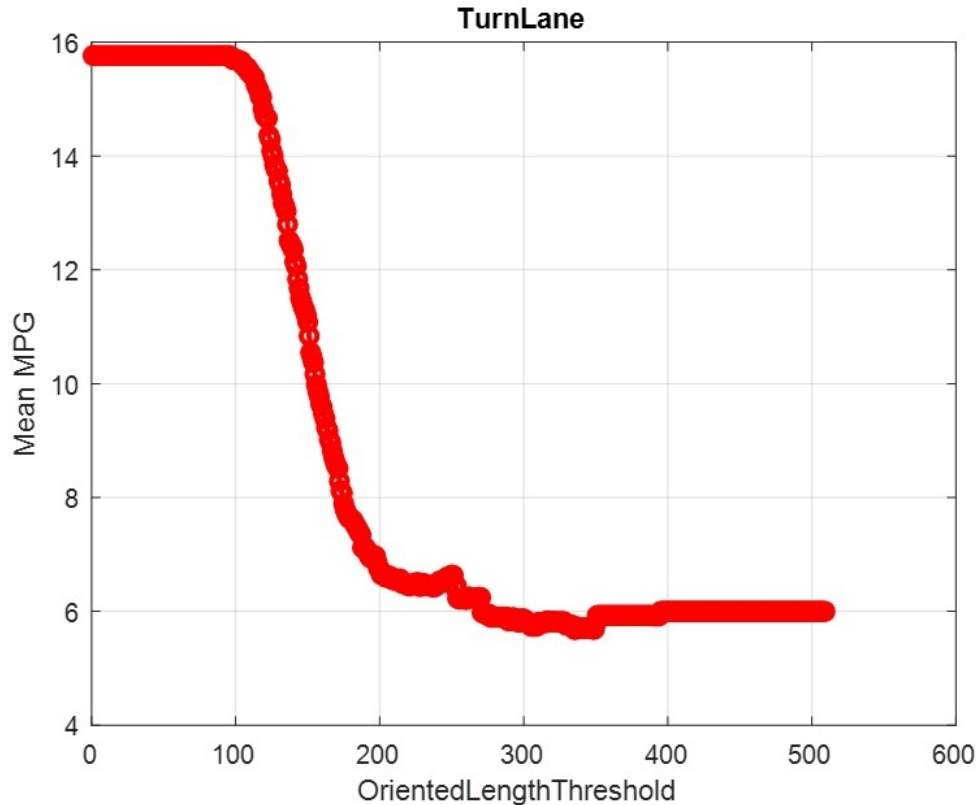
Figure 15. Average fuel consumption estimate in MPG for vehicles in the "turn lane" based on the oriented bounding box length. From left to right we see that the average fuel consumption of vehicles with an oriented bounding box length greater than 0 has an average MPG estimate of approximately 16 MPG. The size threshold increases, and we begin to omit smaller vehicles which have better fuel economy in general. Vehicles above ~350 pixels have a mean MPG of approximately 6 MPG. As in the "FarLane", case, the best explanation for the difference in the mean MPG at lower oriented lengths may be simply differences in vehicle "demographics" for different lanes.

Our second analysis that uses the bounding box is concerned with discriminating "regular" vehicles from the special vehicle classes of high fuel consumers. To this end, we took the largest 100 "regular" vehicles from each region and assigned them to a single class and then attempted to discriminate between the large WGBS classes (18Wheeler, Bus, MultiAxle, DeliveryVan, and Chevrolet Express Bus). As the data set has a limited number of examples of Bus, we elected to combine this class with the 18Wheeler class. Three experiments were conducted, one on each of the heat map regions (NearLane, FarLane, TurnLane). The overall data set was reduced to 60 random examples of each class (18WheelerBus, MultiAxle, DeliveryVan, Chevrolet Express Bus) and 60 random examples from the 100 largest examples of the "regular" vehicle class. A pre-trained convolutional neural network based on the MobilenetV2 [Sandler, 2018] topology was utilized, except that the output of the last fully connected layer was used to create a 1000-dimension feature vector. The data set of 300 vectors was split into five folds, each with a training set of 210 vectors evenly divided between the classes and a testing set of 90 vectors. This was then used to train and test a classifier using an error-correcting output code model [Fürnkranz, 2002] for multiple classes with support vector machines. After each set of folds was completed, a new set was generated using a new random set of examples, and the process repeated 10 times. The overall performance for each lane was approximately 86% regardless of the lane of traffic, indicating that there is a high level of discrimination possible between the largest vehicle types as well as with "regular" vehicles with large bounding boxes. Table 2 through Table 4 shows the confusion matrices, with the known classes in the rows and the estimated classes in the columns.

17

Table 2. Confusion matrix for results in the near lane (accuracy 86%).

	18Wheeler+ Bus (est)	Chevrolet ExpressBus (est)	Delivery Van (est)	Multi Axle (est)	Other Vehicles (est)
18Wheeler+Bus	45.8	0.50	7.2	5.5	1.0
ChevroletExpressBus	0.30	58.3	1.40	0.0	0.0
DeliveryVan	7.90	0.50	45.60	4.6	1.4
MultiAxle	4.50	1.00	3.90	49.7	0.90
Other Vehicles	0.30	0.80	0.90	0.90	57.10

Table 3. Confusion matrix for results in the far lane (accuracy 86%).

	18Wheeler+ Bus (est)	Chevrolet ExpressBus (est)	Delivery Van (est)	Multi Axle (est)	Other Vehicles (est)
18Wheeler+Bus	44.8	0.40	7.6	6.0	1.2
ChevroletExpressBus	0.20	58.5	1.3	0.0	0.0
DeliveryVan	8.70	1.20	46.8	2.9	0.4
MultiAxle	4.20	0.40	4.6	50.2	0.6
Other Vehicles	0.50	0.10	0.30	0.40	58.7

Table 4. Confusion matrix for results in the turn lane (accuracy 86%).

	18Wheeler+ Bus (est)	Chevrolet ExpressBus (est)	Delivery Van (est)	Multi Axle (est)	Other Vehicles (est)
18Wheeler+Bus	47.8	0.590	6.2	4.6	0.6
ChevroletExpressBus	0.30	58.9	1.0	0.1	0.0
DeliveryVan	7.60	0.7	45.4	3.9	2.4
MultiAxle	4.30	0.7	4.5	48.3	2.2
Other Vehicles	0.30	0.2	0.90	1.5	57.1

3.1.6 Variation in Vehicle Types from Fueleconomy.gov

Fine-grained vehicle identification (classifying a vehicle into its make/model and year based on an image) is a difficult problem but has been tackled with some success in the literature and from commercial applications. There is still room for improvement even with constrained images in a real-world environment (such as the GBS images), let alone from the GRIDSMART platform with variation in image quality due to environmental conditions, distances from the road, and simply the level of detail available in these images as discussed earlier. The vehicle dynamics and driving styles are also significant factors, although we speculate that, to a first approximation, the vehicle identification can be a substantial parameter in characterizing fuel consumption. However, a more fundamental problem was discovered by working with the fueleconomy.gov data. and with our work here. In particular, multiple variants of the same visual vehicle type can have significantly different fuel economies. This is intuitively obvious from the most extreme comparison between conventional fuels and hybrid/electric versions of the same vehicle, or even variants such as engine size, diesel vs conventional fuel, etc.

For the vehicle types identified in this data set (and excluding the WGBS classes), Figure 16 shows the standard deviation of combined (highway and city) fuel economy ratings for vehicles that fall under the

class indicated, assuming all variants are equally likely. Vehicles variants that involved unconventional fuels were omitted by inspecting the value of the "atvType" field from fuelecomony.gov. The omitted fuel types were Bifuel (CNG), Bifuel (LPG), CNG, and EV. However we note that their visual characteristics make the problem even more difficult (i.e., telling the difference between an electrified Smart ForTwo and a conventional vehicle is likely impossible from a visual image, without being able to read the script on the vehicle itself, or the presence of a gas tank access door). Based on this finding, we note that several vehicle types have a standard deviation in their fuel consumption estimate in excess of 3 MPG.
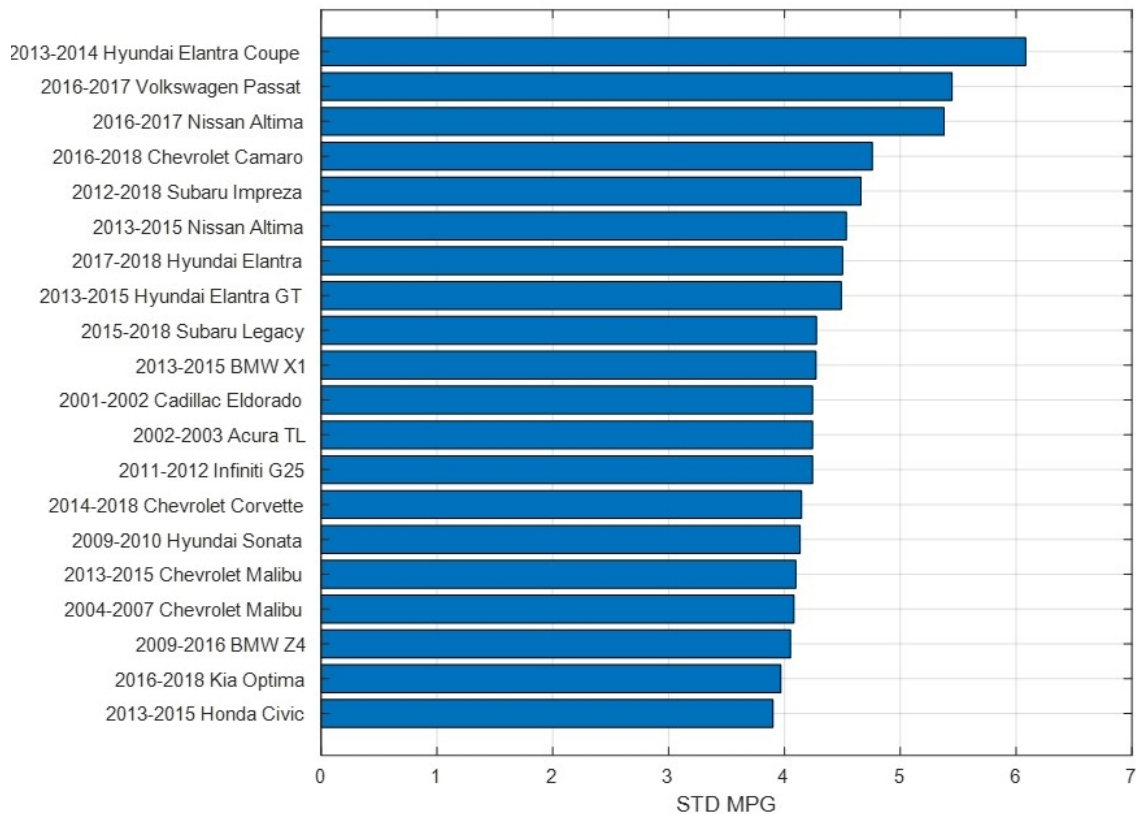
Figure 16. Variation within different variants of the same vehicle class identified in the data set. These variants are due to hybrid versions, different engine sizes, different wheel drive options, etc. The 20 classes with the highest variance are shown here.

3.1.7 Other Data Sets

As a final technical step, we define three other data sets of interest in this work. The first two directly target the concept of "teaching cameras to estimate fuel economy". Image data sets for classification are typically very large, especially for state-of-the-art machine learning/deep learning approaches. Therefore, we had some concern that we would not be able to obtain significant data from the GRIDSMART view. We used two contingencies to estimate vehicle classification performance for our simulations. First, we used a table of vehicle types and the length and width estimates of the vehicles to try and estimate vehicle fuel efficiency using a linear regression model. Second, we used an open-source database of vehicle images to estimate classification performance and its impact on fuel efficiency estimation. (Note that for our initial analysis we did not include large commercial trucks, which we have shown have a definite impact on the system performance.) The estimated fuel economy with this model and perfect measurements is a root mean square (RMS) error of 2.85 MPG, but conversations with GRIDSMART as

well as our findings with segmentation masks indicated that there can be substantial error in such a measurement from any computer vision platform. We found that the regression model rapidly degrades with small measurement error; even a 500 mm mean error degrades the RMS error to approximately 5.8 MPG. Therefore, we expect the utility of a measurement-based system will largely be found in discriminating commercial trucks (particularly 18-wheelers) from passenger vehicles. We believe our complete data set has shown such an effect.

Our second effort in this area used the data set from Gebru et al. (2017). The data set contains over 2500 fine-grained classes of vehicle make/model and year and also includes fuel economy estimates. As mentioned previously, this data set consists of photographs scraped from the worldwide web and are virtually all taken from the "ground view". Nevertheless, we believe it serves as a good estimate for what may be possible with a full data set from the GRIDSMART vantage. We retrained a convolutional neural network based on the Alexnet topology [Krizhevsky et al., 2012] to act as a vehicle make/model classifier. This was inspired by the example of Gebru et al. (2017), which served as a good baseline for the exercise. We trained using 70% training data, 15% validation data, and 15% testing data and evaluated our performance on the test data set aside. We also degraded the image resolution to simulate actual degradation of the image quality from the GRIDSMART imager at ORNL, at ranges of 0 meters, 20 meters, 40 meters, and 60 meters. Finally, we used the classifier to estimate fuel efficiency visually by assuming that if we successfully identified the make and model of the vehicle, our error was 0 MPG; otherwise, we used the erroneous classification as the MPG and measured the RMS error between this estimate and the actual value. The results are summarized in Table 5. An example of a tracked vehicle from 60 m to the stoplight is shown in Figure 3.

Table 5. Estimates of fuel economy using CNN Baseline Model.

Range to Vehicle (meters)	Fine-grained Classifier Accuracy	RMS MPG Error
0 meters	33%	3.5
20 meters	16%	5.1
40 meters	3%	6.7
60 meters	1%	10.0

This data set was used with the Multi-Node Evolutionary Neural Networks for Deep Learning (MENNDL) [Young et al., 2015] tool on ORNL's TITAN to attempt to evolve a better CNN for classifying the data set. Using the body styles available in the data set, MENNDL evolved a CNN which yielded a classification accuracy of 60% on body styles. Previous analysis showed that the body styles do not map as well as expected to MPG estimates, but the utility of MENNDL for this application was proven in these experiments and could be an excellent topic for follow-on work as described later in this document.

The final data set was more intended for use in the simulation work; nevertheless, we include it here to serve as a natural transition into that focus area. The data set was not ultimately utilized and consists of a set of time series data from drivers who passed through intersections in the Second Strategic Highway Research Project (SHRP-2) naturalistic driving study data [Campbell, 2012]. Over 3000 drivers participated in this study, which lasted up to 2 years. The drivers used their personal vehicles which were outfitted with a custom data acquisition system that acquired video data, radar data, and a variety of other sensor data. The goal of the intersection data was to create models of vehicle consumption from real-world drivers for the purposes of creating a custom vehicle simulation. However, as will be described in the next section, we eventually elected to use an open-source simulation package which included a fuel consumption estimate. We still believe there is utility in the data set, however, which contains data from

approximately 25,000 traversals through intersections that were randomly sampled from a complete set of intersection traversals. The data set [SHRP2, 2019] can be downloaded with proper permission and data usage license agreements from the Virginia Tech Transportation Institute, who are the stewards of the SHRP2 data set [InSight, 2018].

3.2   SIMULATION TECHNICAL FOCUS

Our second focus was "teaching a grid of cameras to improve traffic mobility and fuel economy". We note that the GRIDSMART cameras and processing in and of themselves do not control the intersection light timing; rather, they provide a sensor signal that the traffic controller utilizes. However, our simulation goal was to determine the feasibility of traffic control informed by GRIDSMART technology based on our data analysis in the preceding section.

3.2.1   Reinforcement Learning Approach

Reinforcement learning (RL) [Sutton, 1998] seeks to determine the best action to take given a particular environment. RL has been extensively used in game playing applications; one of the early successes was an application in backgammon [Tesauro, 1995] where the environment is the current game board and the action is the move. RL networks typically accept a representation of the environment known as a "state." A network for playing backgammon, for example, might use the location of all the pieces on the game board as the state. Alternatively, it might use an image of the board as the state. Sometimes the state can include historical information, such as the location of pieces over the last several turns.

The "state-space" of a network is the set of all possible states. A common problem in RL is the establishment of a state-space that captures all relevant information without being overly complex; even simple problems can rapidly develop into a large state-space, where many different possible environmental situations can occur. Another problem in RL is the establishment of the "reward" structure, which is how the RL algorithm learns the best actions to take for a given environmental state. For problems with limited time spans, the reward can simply be a metric of success (i.e., you are rewarded if you win the game or alternately penalized if you lose the game). However, if there are longer time spans under consideration, such a reward may be too weak to successfully allow learning, since the network would need to accurately predict the environmental state many time steps in the future. Another issue of note is the learning process where "exploration" is traded off with "exploitation". "Exploration" refers to randomly attempting actions that may not have any short-term reward but may ultimately pay off in the end; while "exploitation" refers to taking the best action in a given environment based on what has already been learned. Finding optimal or near-optimal solutions to RL problems with large state-spaces and various reward options requires applying a great deal of computation to a large number of candidate networks. These complex problems, particularly when taking advantage of recent advancements in RL that use deep networks for environment representation and state-action mapping [Mnih, 2013], derive significant benefit from a HPC environment.

For our particular effort to use RL to explore energy usage, we note that there are multiple papers in the open literature concerned with applying RL to traffic control as well as energy usage [Yau, 2017; Al-Islam, 2018]. In our case we sought to simulate large grids, targeting 25×25 intersections, but we were unable to do so in the limited time allotted for this project. However, we were able to complete smaller simulations that utilized a visual-informed policy akin to the signals obtained from GRIDSMART cameras. In our approach we used a deep network to generate the traffic control changes, but we started with the assumption that a GRIDSMART-like system would operate as sensor and include "edge computing" to deliver a fuel consumption estimate (Figure 17). Much of this edge computing is already done on the GRIDSMART platform, and we have shown the utility of such computing in the data section of this report as well as the potential for more discrimination through the use of object classification. We

also believed such an approach would be more tractable in our limited project scope and may also be more generalizable, since we start with visual-compute models that perform vehicle segmentation and rudimentary classification.
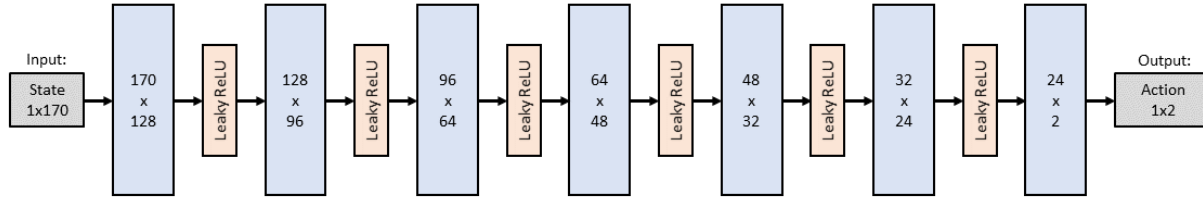

Figure 17. DeepQ network used for simulations. The network is largely a sequence of fully connected layers of neurons and rectified linear unit (ReLU) activation units.

3.2.2   Simulation Approach

Our simulation efforts required a traffic simulator with easy implementation of visual camera sensing, including resolution issues, vehicle classification performance, and control as well as providing methods to estimate fuel consumption and ease of mobility. A method with cluster parallelization capabilities was also desired.

We performed some initial work in FY 2018. In particular, we attempted to begin a custom simulation package with the understanding that it would offer more flexibility, given the limited scope of the project, the team's relative lack of experience in transportation applications, and resource limitations with existing packages. However, additional resource issues arose, and we eventually identified a new team member who considered three approaches in early FY 2019 as possible options for our development: a cell transmission model, an open-source traffic simulator, and the custom simulator approach. The cell transmission model seemed promising but too simplistic to allow us to model a camera sensing for intersections. There was considerable attention paid to the trade-offs between a custom platform and adopting a pre-existing toolset. The custom simulator approach was reconsidered, and a set of "user stories" were developed to improve the development process; however, in parallel with this the open-source package, Simulation of Urban MObility (SUMO) was obtained [Behrisch, 2011]. After some initial startup issues, SUMO became the preferred method due to its scientific basis, history in transportation studies, and flexibility. Indeed, later in the project the SUMO development effort was successfully passed to an additional team member with experience in machine learning. This team member was able to demonstrate that, given fuel use estimates similar to those that can be estimated from GRIDSMART images, RL is capable of reducing overall fuel consumption by intelligently controlling the phase of traffic lights.

SUMO features a traffic control interface (TraCI) which allows control over how the simulation functions with respect to traffic conditions. Our team was also successful in operating this functionality to enable the simulation of different traffic control policies. We also integrated some machine learning packages, specifically Keras/Tensorflow (2019), to implement our reinforcement learning algorithms. Finally, we were also able to install and run SUMO on the ORNL TITAN HPC platform and generated some timing information regarding its application. We were able to run a 25×25 intersection grid in SUMO on TITAN in roughly ½ real time (the simulation of 1 hour of traffic required 1800 seconds to run). We did explore some parameter trade-offs that would increase the speed, specifically how often the TraCI package was called for control. We note that the timing on an individual Titan node is not as fast as modern

workstations (Titan has reached its End of life and will be decommissioned in 2019). The real benefit of HPC is massive computer parallelism and inter-process communications which must be carefully optimized to extract the maximum compute power from the platform. As a final note, modern leadership class HPC platforms provide significant benefits by leveraging more powerful nodes (a node on ORNL's new Summit supercomputer is roughly 29X times faster than a single Titan), so any future work may be able to leverage these advantages.

3.2.3 Results

The SUMO simulation environment used a "visual model" based on our initial findings with the GRIDSMART camera. In particular, we limited the sensing space to 60 m around the intersection and used the MPG RMS errors found in Table 5 for an "error" case of fuel economy. We also used a "perfect" or no-error comparison where the vehicle was assumed to be perfectly identified. The grid traffic light distances were set at 500 m apart. The traffic distribution was 50% buses/trucks and 50% passenger vehicles. Buses and trucks traveled north or south, and passenger vehicles traveled in any direction. This skewed distribution was utilized to help verify that the simulation and RL model were learning information from the environment to achieve our goals.

Four different policies were tested: (1) a control policy based on a fixed timer (30 seconds green and 6 seconds yellow); (2) a heuristic policy where the fuel consumption was computed in each lane with the visual model, and then the phase was changed if the highest consuming lane had a red light; (3) a RL policy that uses fuel usage estimates from vehicles within 60 m of the traffic light (orange regions shown in Figure 18); and (4) a RL policy that uses fuel usage estimates from vehicles with 60 m of the traffic light and also vehicles within 60 m of adjacent traffic lights (orange and blue regions shown in Figure 18. For the RL policies, the state was defined to consist of the following:

- The traffic light's current phase

- The number of seconds the light has been in the current phase

- Fuel usage estimates from each orange region shown in Figure 20 each second for the last 3 seconds

- For the second RL policy only, fuel usage estimates from each blue section shown in Figure 18 each second for the last 35 seconds

During training, RL networks seek to maximize some reward function. Although rewards are often formulated as a positive value, such as a large positive value if a network that is being trained to play a game wins the game, we use penalties with negative values. The reward for our networks has two elements: (a) a penalty that is proportional to the amount of fuel used by vehicles that are stopped at the traffic light, and (b) a penalty that is applied if the network tries to change a yellow light.

The results of the simulations are shown in Figure 19 through Figure 28. (We first want to note that these are based on the best simulations we could perform, but we certainly believe more simulations are needed for additional confirmation of results.) The dense simulations generate vehicles every 1–4 seconds until a total of 500 vehicles are utilized, and all results are averaged over 10 simulations. The total fuel usage in gallons was provided by the SUMO output and shows that the visual policies have better performance (note that the x axis starts at 400 gallons). There appears to be little difference between the "error" and "no error" visual models, suggesting that the classifier accuracy does not need to be highly optimized to provide benefits. All visual policies outperform the control strategy for vehicle stoppage time; this is intuitively obvious because the control policy is simply timer based and effectively has no sensing at all.

A better comparison may be to use a simple occupancy sensor to compare, although we still believe the visual policies will function better especially for fuel consumption.
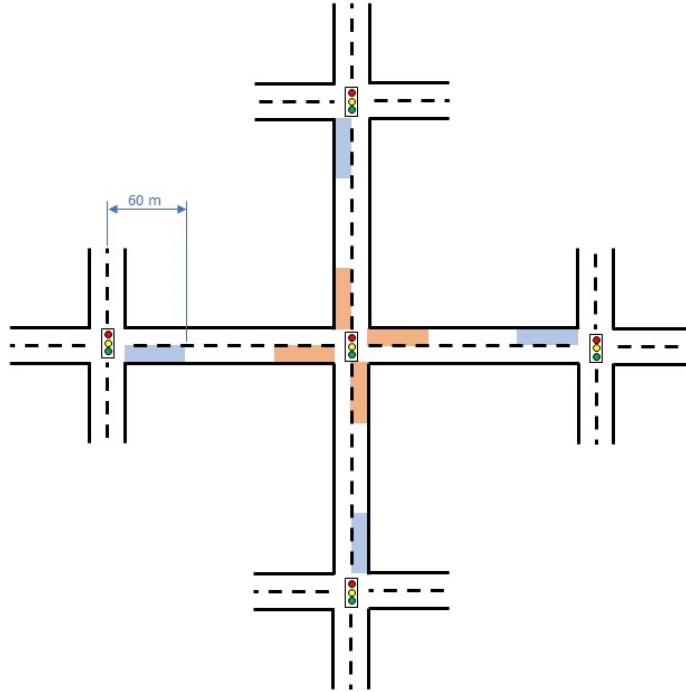

Figure 18. Diagram depicting states at intersection and adjacent lights.

The scatter plots in Figure 21 through Figure 28 show more information about the processing. There are two distinct distributions; on the left we see the passenger vehicles and the right, the larger "gas guzzling" busses and trucks. The control policy stops vehicles up to approximately two minutes with no preference; note that the right side distribution especially is slanted because the fuel consumption rate goes down as the vehicle is stopped longer (reflecting the lower fuel consumption of stopped vehicles). However, the visual policies show a distinct flattening of the wait times for larger vehicles, which suggests the intersection is learning to allow them to pass without stopping, which saves energy. These effects are particularly noticeable in the sparse experiments, where vehicles were generated every 10 seconds. One unexpected result, which can be seen in Figure 23 ("Total stopped vehicle tine in simulation for dense traffic experiments") is that the non-adjacent Deep Q RL model produces longer average stopping times than the heuristic policy. It is important to note, however, that despite the extra stopping time, the model still results in less fuel use than the heuristic policy. In the future, we may be able to minimize stopping time by incorporating it into the RL network's reward function, and penalizing exceptionally long stopping times.
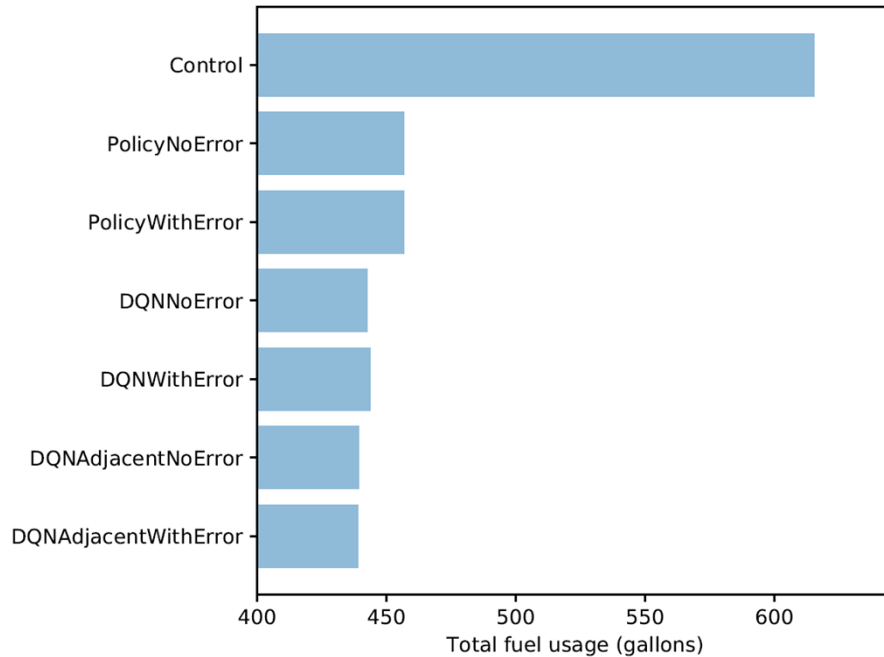
Figure 19. Fuel usage for different policies under the dense traffic experiments. The visual policies outperform the simple timing policy, with the RL methods performing slightly better. Note that the error in the calculation has little effect, which is likely due to the higher ability to identify truck / bus traffic.
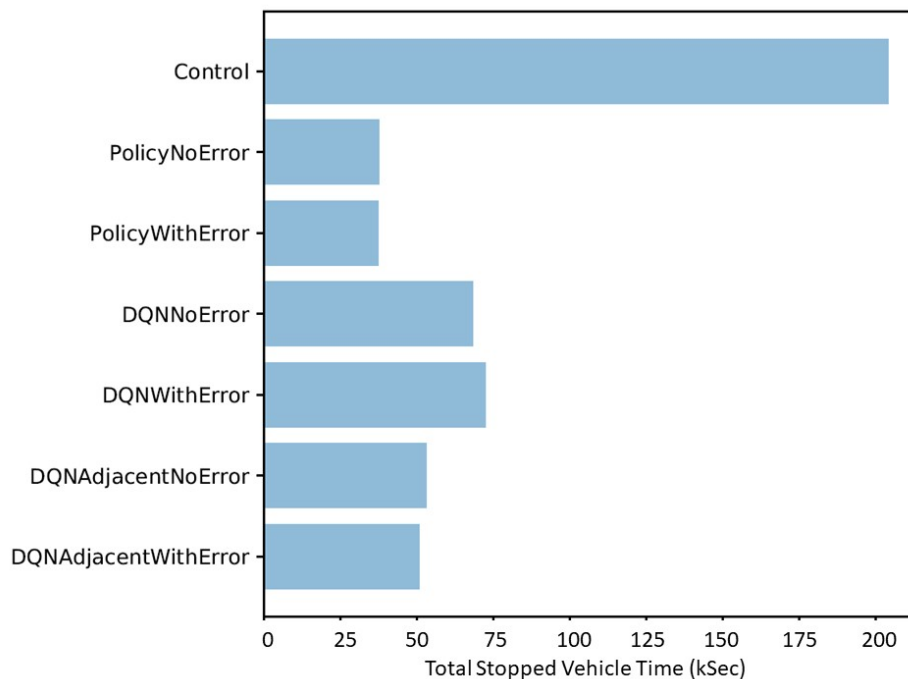

Figure 20. Total stopped vehicle time in simulation for dense traffic experiments. There were 10 different experiments conducted with new vehicles entering the simulation every 1–4 seconds for 5000 total vehicles. The control policy stopped time is roughly 40 seconds per vehicle which aligns well with the policy of 30 seconds/green light.
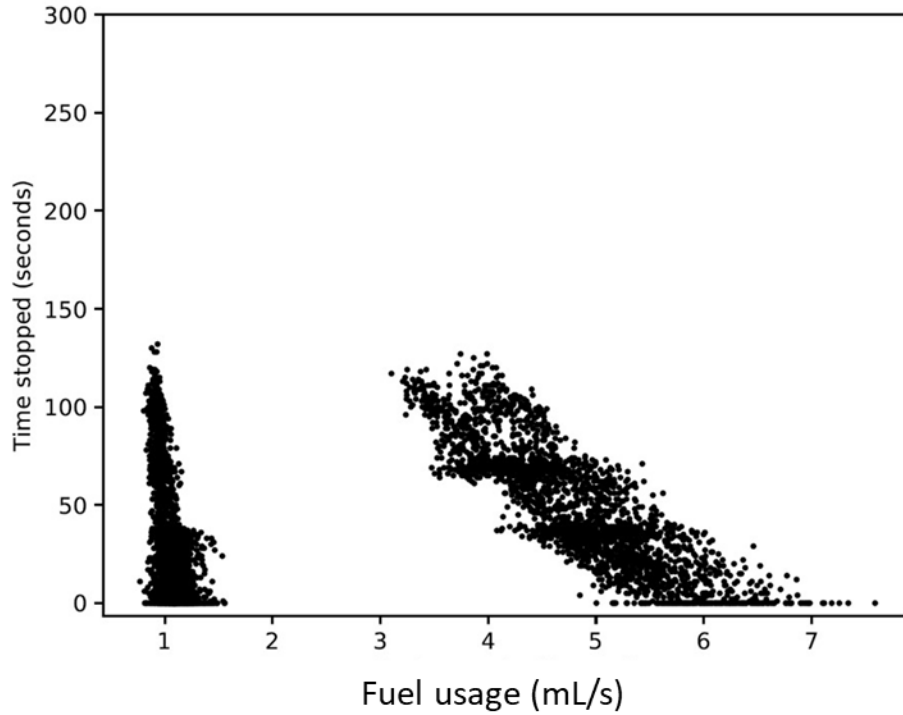
Figure 21. Control policy with dense traffic. The distribution on the left is the passenger vehicles and that on the right is busses and trucks. The slant effect, particularly on the right, is caused by the decrease in the vehicle fuel consumption rate as vehicles are stopped for longer periods of time.
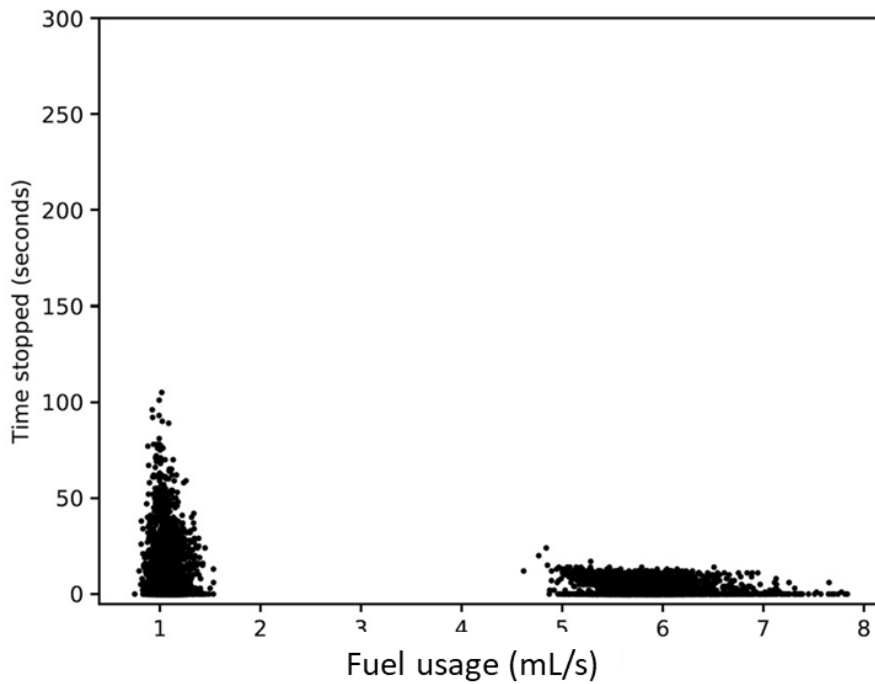

Figure 22. Heuristic vision-based policy with dense traffic. The "flattening" of the right distribution is caused by reducing the amount of time busses and trucks are stopped.
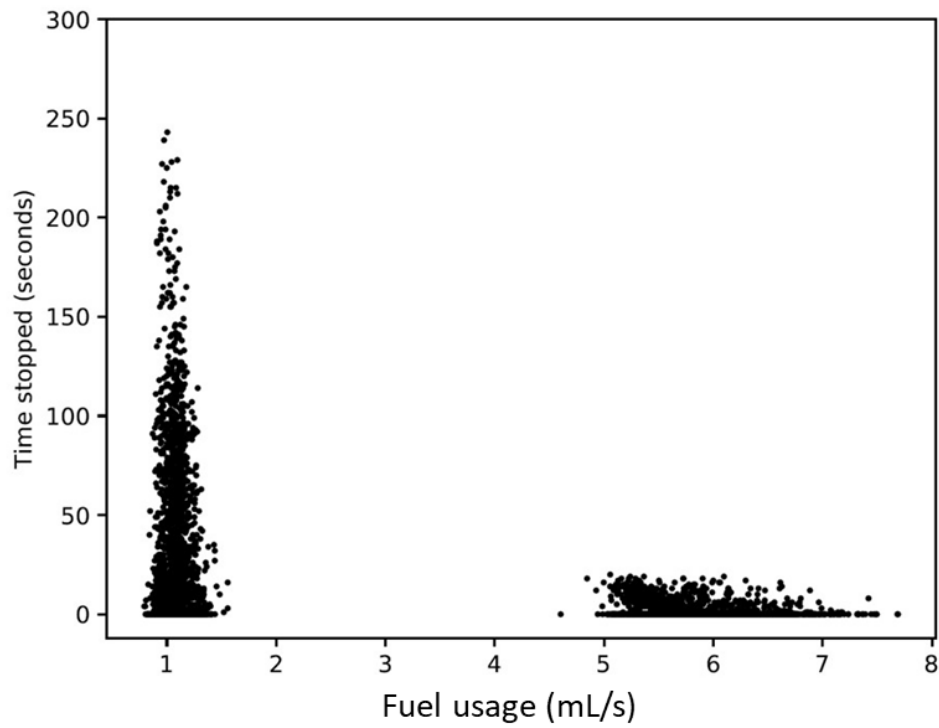
26

Figure 23. DeepQ policy with dense traffic. This model also has a flattening of the right distribution but has longer stopped times for uncertain reasons, possibly due to shorter training times.
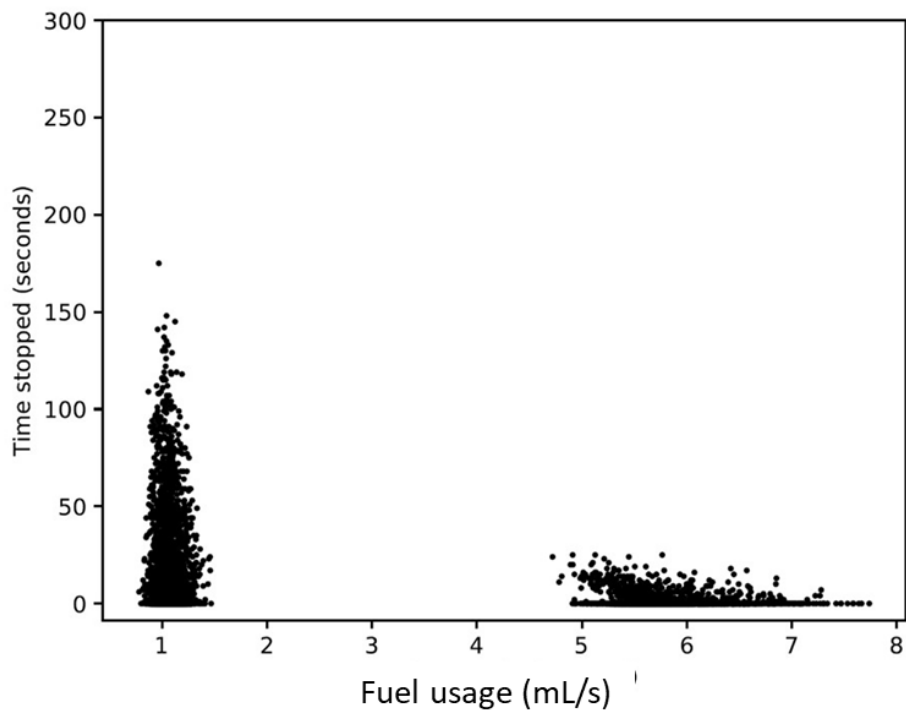

Figure 24. DeepQ Adjacent with dense traffic.
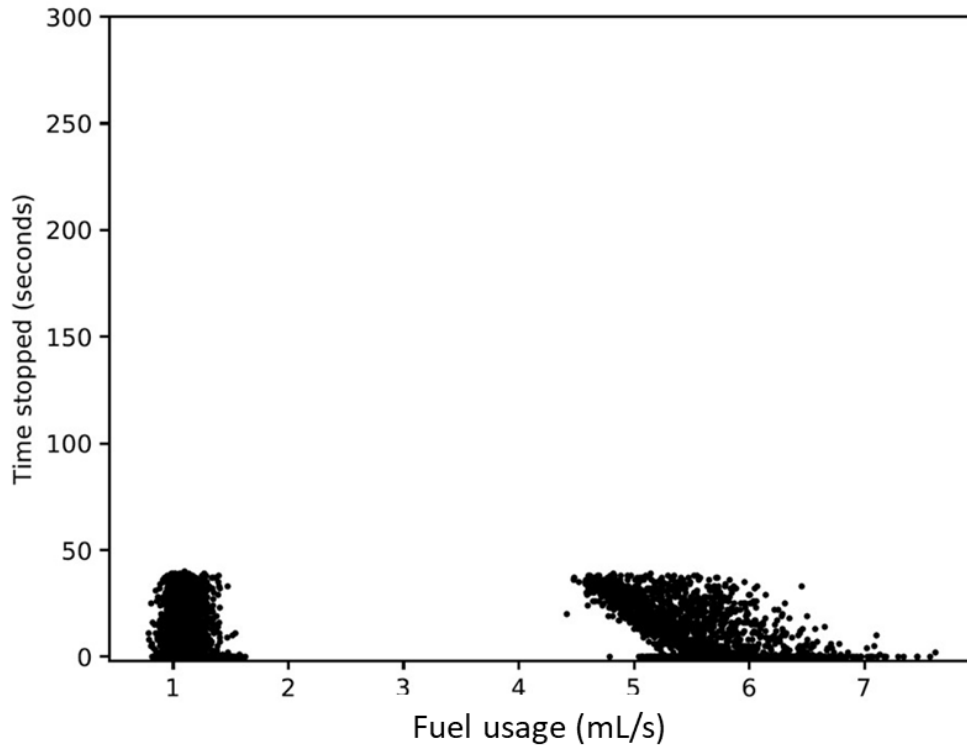
Figure 25. Control policy with sparse traffic.
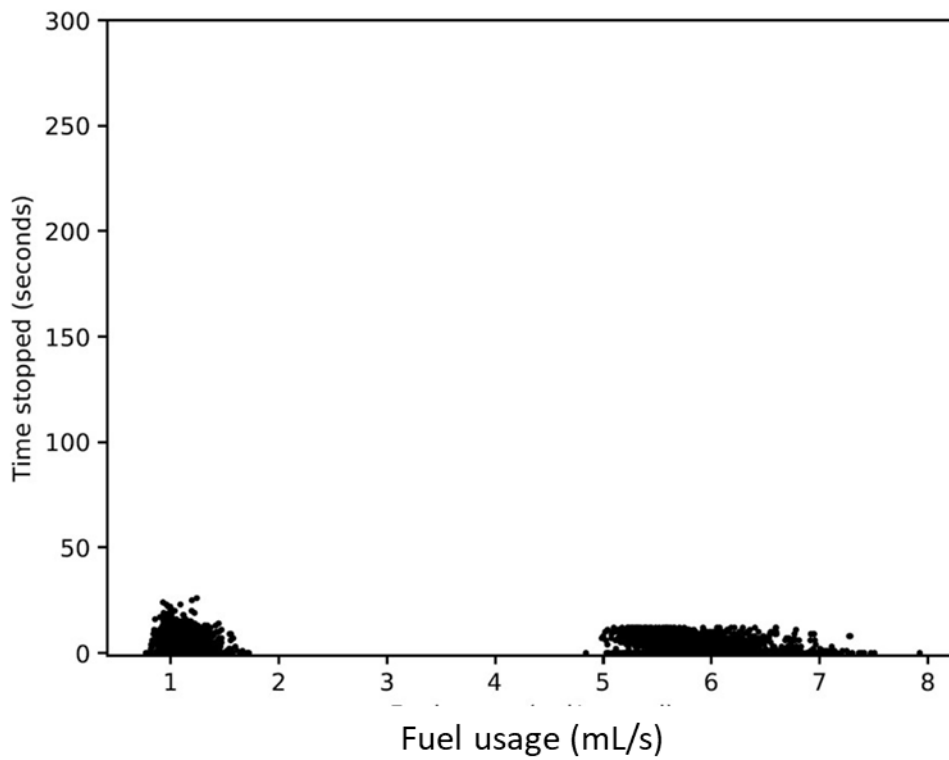
Figure 26. Heuristic vision policy with sparse traffic, showing better stopped time than the control policy along with some flattening of the bus/truck distribution on the right side of the plot.

Figure 27. DeepQ policy with sparse traffic, with a rather long stopped time distribution for smaller vehicles.

Figure 28. DeepQ + Adjacent for sparse traffic. This case shows the most extreme "flattening" of the right-side distribution of traffic.

4.   GRIDSMART CONTRIBUTIONS

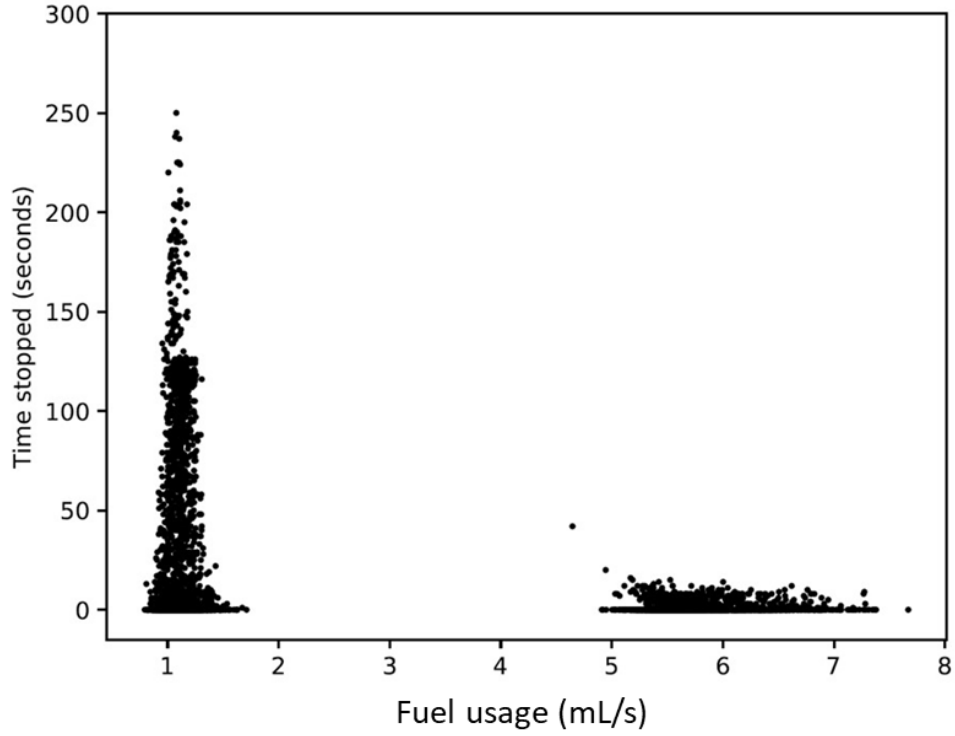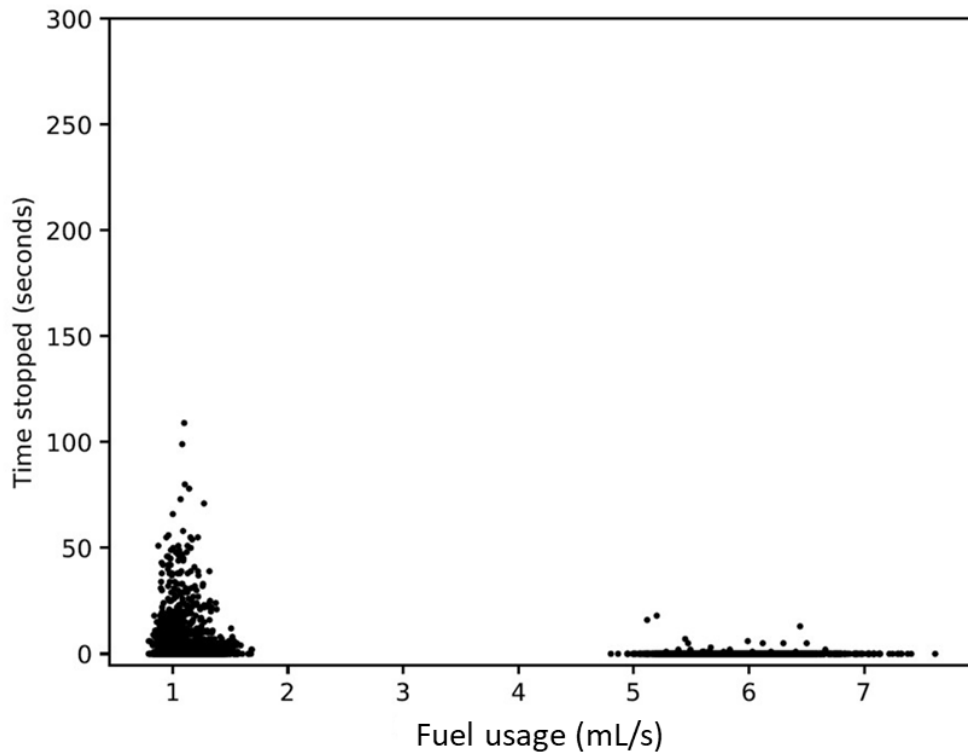During both focused efforts, GRIDSMART personnel provided invaluable information regarding their deployments, algorithmic approach, constraints, and business plans/practices. They trained ORNL personnel on proper data acquisition techniques using GRIDSMART camera technology, in particular on best practices to acquire data without interfering with the core GRIDSMART function of providing real-time information on the traffic situation to the light controllers. They provided disk specifications for data acquisition and Application Programming Interface (API) information for data analysis and review purposes. They worked closely with ORNL personnel to implement a Network Time Protocol (NTP) time synchronization solution, without which the data collection effort would have been impossible. In the process of acquiring data and understanding the capabilities of the effort, GRIDSMART personnel identified a problem with one of the controllers (a failing cooling fan) and initiated a repair, potentially preventing a more serious failure.

5.   SUBJECT INVENTIONS (AS DEFINED IN THE CRADA)

There are no subject inventions to date. Note, however, that a publicly available data set of vehicles was created from the GRIDSMART "point of view", with varying levels of ground truth (from vehicle make, model, and year range to broader classifications such as Bus or MultiAxle) and will be published subject to ORNL and GRIDSMART approval (https://www.ornl.gov/group/isml/datasets/oovd) In addition a draft publication has been prepared for submission to relevant conferences or journals (Karnowski et al, "Estimating Vehicle Fuel Economy from Overhead Camera Imagery and Application for Traffic Control")

6.   COMMERCIALIZATION POSSIBILITIES

As this CRADA was a seedling project, the commercialization possibilities are largely speculative and rely on potential partnerships beyond ORNL and GRIDSMART. GRIDSMART does not manufacture traffic control systems; rather, their products are sensors for traffic control and data products for transportation system planning and analysis. However, the project proved the concept of building ground-truth data sets using GRIDSMART cameras and ORNL technology with the ground-based sensor, along with the commercial application for vehicle classification. While more manual data hygiene was required than was originally expected, the value of the data set to provide better traffic analytics was illustrated along with the potential to make a rough visual estimate of fuel consumption from both the vehicle-oriented bounding box size and the acquired data set. The potential use of GRIDSMART imagery for adaptive traffic control was also illustrated through the SUMO simulations, with improvements seen in the use of visual information as a heuristic policy and through the Deep-Q learning algorithms (with information both at the localized intersection and with information from adjacent intersections). However, we note that GRIDSMART has challenging requirements for its real-world deployments that must meet a variety of environmental and power conditions, which put constraints on computational resources and other implementation details.

7.   PLANS FOR FUTURE COLLABORATIONS

There are multiple potential follow-on activities from this work that would extend the impact of this proof-of-concept project. First, extending the simulation packages to larger simulations with more advanced control is an obvious follow-on activity. A municipal partner would be a good choice for this activity, particularly one with multiple intersections with visual sensing. Second, a larger data set could be obtained, as the GBS is more advanced at this point of development. Although a better vehicle

classification system would be beneficial, other information such as automated license plate readers or higher resolution vehicle captures would increase the ability to collect ground-truth data. (All such work would require proper Institutional Review Board approvals.) A true field study could be performed with additional research & development (Figure 29) but would require both GRIDSMART collaborations as well as a traffic controller partner; as an example, ORNL could serve as a test site if proper permissions were obtained from authorities. The SNS intersection feeds directly to the Melton Valley intersection, and two cameras could be placed to the west on Bethel Valley Road and to the south to capture vehicles moving toward the Melton Valley intersection. This arrangement would simulate a grid environment as a small-scale testbed. A communications pathway would be required, and the existing cameras would require modifications or interfacing hardware built to utilize the camera images in real time. Finally, the tools developed for this project can increase the situational awareness in a transportation network to augment and enhance projects developing digital twins and data analytics, particularly for networks that already utilize GRIDSMART technology and products.
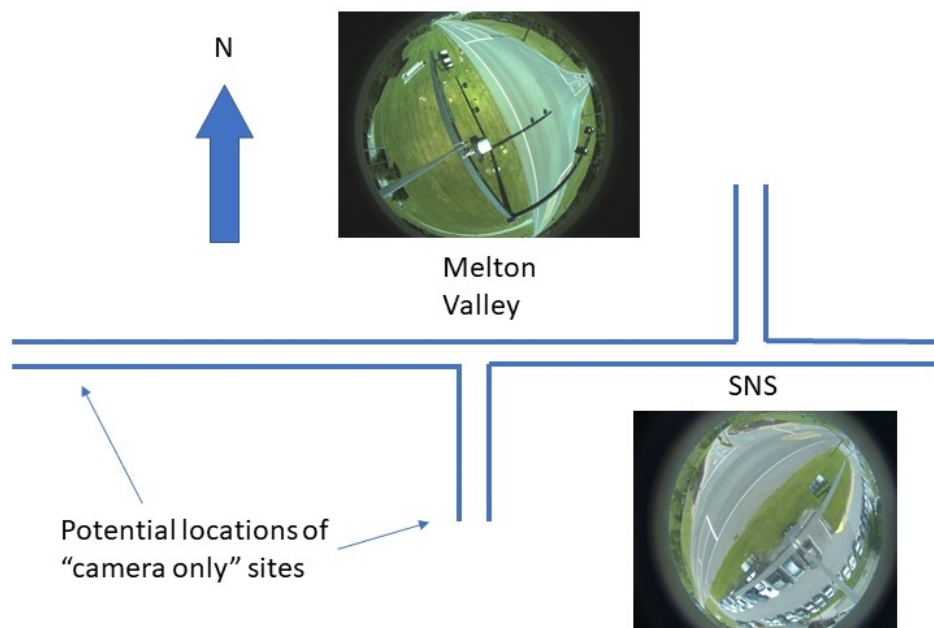

Figure 29. Conceptual diagram of test site for visual/RL-based intersection control. The direct path from SNS to Melton Valley could be augmented with camera-only sites on adjacent roads that would feed visual information to the controller at Melton Valley. This idea is presented as a conceptual concept only and would require multiple collaborators, additional research & development, and appropriate authorities prior to testing.

From the standpoint of HPC, the MENNDL tool developed at ORNL [Young et al., 2015] has enormous potential in follow-on activity. MENNDL could be used to evolve better deep-Q networks for control by using a massively parallel simulation where each node could run SUMO in conjunction with a different deep-Q topology and reward structure. Genetic algorithms could breed better networks and reward structures over time. MENNDL could also be used to produce better fine-grained vehicle classifiers (Figure 30). Our efforts on MENNDL for this project were limited to body style vehicle classifiers, and the initial results were promising. Another option is to use MENNDL to create hardware-constrained networks that may be more amenable to real-world deployments. In this application, MENNDL would seek to improve classifier performance but also use other criteria such as network size, number of parameters, latency, etc., or even run without GPUs which could be done in principle on native hardware within the HPC.
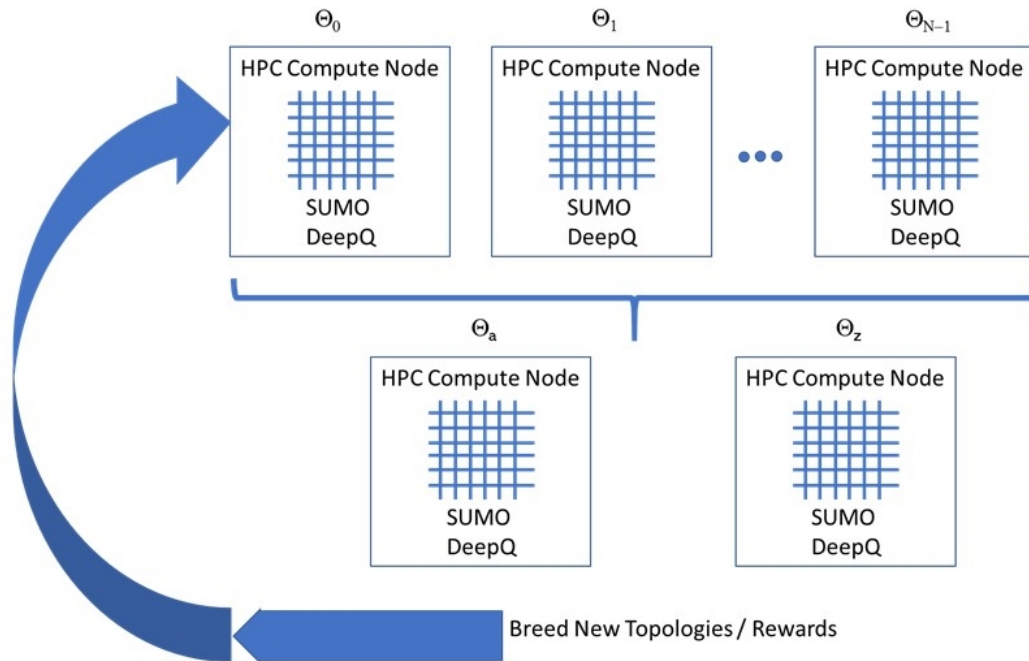
Figure 30. Conceptualization of using HPC to create new network topologies, reward structures, etc., to improve traffic control for energy and mobility. In each node, a different SUMO simulation is performed with different DeepQ topologies, rewards, etc., and the best performers are used to create a new generation of parameters which are run until an optimal policy is achieved. MENNDL could be used as the platform for this application.

8. CONCLUSIONS

Improved transportation efficiency is vital to America's economic progress. The increasing congestion and energy resource requirements of transportation systems for metropolitan areas require research in methods to improve and optimize control methods. This proof-of-concept project was able to show the efficacy of building ground-truth data sets for vehicle classification from overhead traffic cameras, in particular GRIDSMART fisheye lens cameras which exist primarily as sensors for traffic control. The data set includes a computer vision vehicle segmentation algorithm that is similar to the approach of GRIDSMART, which must run in a hardware and environmentally constrained environment. The utility of the data set was shown by allowing classification of large fuel consumers that are physically large as well as showing how classifiers built from the data can add higher levels of vehicle discrimination, with the caveat that our discriminator requires deep learning activations which may require specialized hardware. We also found that visual classification has limitations; as the resolution of the data decreases, visual classification degrades significantly. Furthermore, findings from fueleconomy.gov show that variants of vehicle types (such as alternate fuels, different transmissions, and other factors) add increased uncertainty to vehicle fuel consumption characteristics. (We note that vehicle dynamics add another factor of uncertainty.) We showed, through simulations using SUMO, that a deep-Q network coupled with visual sensing emulative of GRIDSMART capabilities could improve transportation efficiency. Although we were not able to extend to larger grid systems due to resource constraints, we did show that information from adjacent intersections could produce a better control policy. Much of the information required to enact these techniques is virtually "free" as it is already part of GRIDSMART's analytics products or could be enacted with algorithms available today (albeit subject to hardware constraints based

on GRIDSMART and transportation specifications). In the future, as vehicles become more fuel efficient through better technology, the gains obtained by cameras with low detection resolution may be more difficult to realize and may rely more on vehicle self-identification. We speculate that such self-identification may be subject to "hacking" either intentional or malicious, and therefore sensing may still play a role to confirm identifications for enforcement purposes, but clearly such work is beyond the scope of this project. Nevertheless, the role of sensing will be significant in intelligent transportation systems of the near and extended future for some time to come.

9.   ACKNOWLEDGMENTS

10.   REFERENCES

Argonne National Laboratory. 2019. Energy Systems Division, Reducing Vehicle Idling, https://www.anl.gov/es/reducing-vehicle-idling.

Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. 2017. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." *IEEE transactions on pattern analysis and machine intelligence,* 39(12), 2481–2495 (2017).

Bay, Herbert, Tinne Tuytelaars, and Luc Van Gool. 2006. "Surf: Speeded up robust features." European conference on computer vision. Springer, Berlin, Heidelberg, 2006.

Behrisch, Michael, et al. 2011."SUMO–simulation of urban mobility: an overview." Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation. ThinkMind, 2011.

Berry, Irene Michelle. 2010. *The effects of driving style and vehicle performance on the real-world fuel consumption of US light-duty vehicles*. Dissertation. Massachusetts Institute of Technology, 2010.

Campbell, Kenneth L. 2012. "The SHRP 2 naturalistic driving study: Addressing driver performance and behavior in traffic safety." *Tr News* 282 (2012).

Dehghan, A., S. Z. Masood, G. Shu, E. G. Ortiz. 2017. "View independent vehicle make, model and color recognition using convolutional neural network." In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017.

EERE. 2019a. About the Office of Energy Efficiency and Renewable Energy, https://www.energy.gov/eere/about-office-energy-efficiency-and-renewable-energy

EERE. 2019b. Energy Efficient Mobility Systems, Vehicle Technologies Office, https://www.energy.gov/eere/vehicles/energy-efficient-mobility-systems

EERE. 2019c. Download Fuel Economy Data, obtained in March 2019; https://www.fueleconomy.gov/feg/download.shtml.

EERE. 2019d. Alternate fuels data center site, https://afdc.energy.gov/data/10310.

Fürnkranz, Johannes. 2002."Round robin classification." *Journal of Machine Learning Research* 2, 721–747, March 2002.

Gebru, Timnit, Jonathan Krause, Yilun Wang, Duyun Chen, Jia Deng, Erez Lieberman Aiden, and Li Fei-Fei. 2017. "Using Deep Learning and Google Street View to Estimate the Demographic Makeup of the US." arXiv preprint arXiv:1702.06683 (2017).

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville. 2016. *Deep Learning*. MIT Press, 2016.

InSight. 2018. InSight Data Access Website. https://insight.shrp2nds.us/.

Al Islam, SMA Bin, HM Abdul Aziz, Hong Wang, and Stanley E. Young. "Minimizing energy consumption from connected signalized intersections by reinforcement learning." In 2018 21st International Conference on Intelligent Transportation Systems (ITSC), pp. 1870-1875. IEEE, 2018.

Krause, Jonathan, et al. 2013. "3d object representations for fine-grained categorization." In Proceedings of the IEEE International Conference on Computer Vision Workshops (2013).

Krause, Jonathan, et al. 2015. "Fine-grained recognition without part annotations." In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015).

Krizhevsky, Alex, Ilya Sutskever, and Geoffrey E. Hinton. 2012. "Imagenet classification with deep convolutional neural networks." In Advances in neural information processing systems, pp. 1097–1105, 2012.

Krizhevsky,A. et al. 2012. "Imagenet classification with deep convolutional neural networks." *Advances in neural information processing systems*, 25(2), 2012.

MathWorks. 2019a. Computer Vision Toolbox. https://www.mathworks.com/products/computer-vision.html

MathWorks. 2019b.vision.ForegroundDetector, https://www.mathworks.com/help/vision/ref/vision.foregrounddetector-system-object.html

Mnih, Volodymyr, et al. 2013. "Playing atari with deep reinforcement learning." arXiv preprint arXiv:1312.5602 (2013).

OpenCV. 2019. OpenCV 4.1 is here. https://opencv.org/

Rakha, Hesham A., et al. 2011. "Virginia tech comprehensive power-based fuel consumption model: model development and testing." *Transportation Research Part D: Transport and Environment*, 16(7), 492–503 (2011).

Sandler, Mark, et al. 2018. "Mobilenetv2: Inverted residuals and linear bottlenecks." In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2018.

SHRP2. 2019. https://doi.org/10.15787/VTT1/4JOEAP.

Sighthound. 2019.Vehicle Recognition Benchmarks, https://www.sighthound.com/technology/vehicle-recognition/benchmarks

Sutton, Richard S., and Andrew G. Barto. 1998. *Introduction to reinforcement learning*. Vol. 135. Cambridge, MIT press, 1998.

TensorFlow. 2019. Keras. https://www.tensorflow.org/guide/keras

Tesauro, Gerald. 1995. "Temporal difference learning and TD-Gammon*." Communications of the ACM,* 38.3, 58-68 (1995).

Yang, L., P. Luo, C. C. Loy, and X. Tang. 2015."A large-scale car dataset for fine-grained categorization and verification." In The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2015.

Yau, Kok-Lim Alvin, et al. 2017."A survey on reinforcement learning models and algorithms for traffic signal control." ACM Computing Surveys (CSUR) 50.3 (2017): 34.

Young, Steven R., Derek C. Rose, Thomas P. Karnowski, Seung-Hwan Lim, and Robert M. Patton. 2015. "Optimizing deep learning hyper-parameters through an evolutionary algorithm." In Proceedings of the Workshop on Machine Learning in High-Performance Computing Environments, p. 4. ACM, 2015.