Estimating Vehicle Fuel Economy from Overhead Camera Imagery and Application for Traffic Control

Thomas Karnowski,[a] Ryan Tokola,[a] Sean Oesch,[a] Matthew Eicholtz, [b] Jeff Price, [c] Tim Gee, [c]
[a]Electrical and Electronics Systems Research Division, Oak Ridge National Laboratory, Oak Ridge, TN 37831
[a] Department of Computer Science, Florida Southern College, Lakeland, FL 33801
[c]GRIDSMART Technologies Inc., Knoxville, TN 37931

Abstract

In this work, we explore the ability to estimate vehicle fuel consumption using imagery from overhead fisheye lens cameras deployed as traffic sensors. We utilize this information to simulate vision-based control of a traffic intersection, with a goal of improving fuel economy with minimal impact to mobility. We introduce the ORNL Overhead Vehicle Data set (OOVD), consisting of a data set of paired, labeled vehicle images from a ground-based camera and an overhead fisheye lens traffic camera. The data set includes segmentation masks based on Gaussian mixture models for vehicle detection. We show the data set utility through three applications: estimation of fuel consumption based on segmentation bounding boxes, vehicle discrimination for vehicles with large bounding boxes, and fine-grained classification on a limited number of vehicle makes and models using a pre-trained set of convolutional neural network models. We compare these results with estimates based on a large open-source data set of web-scraped imagery. Finally, we show the utility of the approach using reinforcement learning in a traffic simulator using the open source Simulation of Urban Mobility (SUMO) package. Our results demonstrate the feasibility of the approach for controlling traffic lights for better fuel efficiency based solely on visual vehicle estimates from commercial, fisheye lens cameras.

Keywords: *computer vision, reinforcement learning, image segmentation, data sets*

** corresponding author's E-mail: karnowskitp@ornl.gov*

Introduction

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.[1] Consequently the increasing energy resource requirements of transportation systems for metropolitan areas require research in methods to improve and optimize control methods. In this work, we explore the ability of commercial overhead-mounted, fisheye lens cameras to provide real-time information to the traffic infrastructure, particularly with respect to fuel consumption. These cameras sense the presence of vehicles at intersections to replace more conventional sensors. The ultimate goal of our work is to leverage these sensing capabilities to improve the energy efficiency aspects of traffic control. Surveys from Department of Energy (DOE) national laboratories estimate that the fuel cost of idling is 6 billion gallons wasted annually[2] which gives a bound on the problem this proposed solution could address.

We seek to determine the degree that overhead traffic cameras can learn to estimate fuel consumption of vehicles in their visual field. We also seek to teach a traffic controller (or a grid of controllers) how to use such information to more efficiently control traffic flow from an energy and mobility standpoint.

This paper is organized as follows. First, to determine the ability of cameras to estimate fuel consumption, we describe a data set acquired at Oak Ridge National Laboratory (ORNL) using overhead cameras with the assistance of higher-resolution, ground-based sensors. We explore use cases with this data set and describe efforts to more thoroughly understand and bound the ability to estimate fuel consumption from visual sensors. Second, we perform some basic simulations that leverage visual models to improve traffic efficiency using open-source simulation packages. We conclude by summarizing our findings and recommend next steps for further development.

Approach

Overhead Vehicle Data Set

Using vehicle imagery to estimate fuel consumption is challenging. We note that fuel consumption models use both vehicle characteristics and dynamic information.[3] In this work we assume the vehicle characteristics are a good first approximation for fuel consumption for two main reasons: first, vehicle dynamics measurements require calibrated cameras which are generally not available in this domain; and second, vehicle visual classification provides insight into traffic characteristics.

We ground our work in the real-world practice of computer vision traffic sensing, and assume our imagery is obtained from a well-engineered system that detects vehicle traffic to replace ground-loop induction sensors. Machine learning for visual object recognition has made great advances with the publishing of large labeled data sets and deep learning algorithms.[4] The technology is ubiquitous but a large problem is the lack of data to customize and train algorithms to attain desired accuracies, particularly regarding real-world conditions (weather and lighting variations).

Therefore, we began with data collection on the ORNL campus to identify vehicle types with a high-resolution image and map these classifications onto the overhead camera view. The resulting data set is dubbed the ORNL Overhead Vehicle Data set (OOVD). (https://www.ornl.gov/project/ornl-overhead-vehicle-dataset-oovd) and leverages a ground-based sensor (GBS) system that captures a relatively high-resolution image of the vehicle of interest simultaneously with an overhead imager. The GBS image is classified by some method (e.g., manual inspection, the use of auxiliary information such as license plates, or machine-learning based object classification), which creates labels for the overhead images. Although there are data sets for fine-grained vehicle classification, they are largely from web-based URLs and therefore are different from the automated "real world" images captured from the GBS and overhead camera.

Data collection consisted of accessing the overhead camera interface unit within the traffic instrumentation cabinet; for the case of overhead cameras available from GRIDSMART, the adopted procedure used an external universal serial bus (USB) drive. A 1 TB drive stores roughly 10 days of data, organized by the hour, with approximately 26,000 images per hour at 7 frames per second. The accompanying ORNL GBS captures multiple images of a target vehicle and uses embedded algorithms to create a projected image where the wheels are aligned from image to image.

There are multiple fine-grained vehicle recognition systems in existence.[5,6,7,8] Consequently, a commercial application was identified to provide vehicle make and model from the high-resolution GBS image. Other labeling methods were used including manual review and matching from the GBS system.

We also sought to emulate an embedded system image processing pipeline using computer-vision-based tools that perform vehicle segmentation in real time. We used the MATLAB Computer Vision Toolbox[9], to detect vehicles via a mixture of Gaussians model. The implementation used a foreground detector 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 image dilation and an estimate of the vehicle location based on an observed trajectory map.

We used the GBS collections as a screening process and only performed vehicle segmentation on the overhead imagery to target a particular GBS image. We identified a range of likely overhead frames that would contain the GBS collections, with a manual selection of the candidate vehicle in the overhead view due to slight timing offsets from synchronization drift between the overhead imagery and the GBS system. Additional data hygiene was required, including the selection of the vehicle of interest, an initial selection of the vehicle direction, and the lane of travel. A second process identified 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)[10]. 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 best frame was saved with information such as the vehicle bounding box, oriented length, and collection time.

In addition to GBS-based collections, there were a number of unique vehicle types that we also identified through a segmentation process. These included larger vehicles that were not identified by 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 are 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.

In the resultant data set, a total of 6,695 vehicles were identified in 685 different vehicle categories. The vehicle distribution is not uniformly distributed, as roughly 150 classes have a single vehicle and some have as many as 238. We 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*. Variation in 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-overhead pair to ensure the classifications seemed correct and that the GBS-overhead images were the same vehicle. However, we must allow for possible errors in the data collection and screening process, and we ask that any errors found by researchers be shared for future corrections.

The data set includes the vehicle segmentation mask, which is the same size as the intersection view. An example image is shown in Figure 1. Finally, the data set also includes an estimated fuel economy value from either the U.S. Department of Energy Fuel Economy archive[11] or the alternate fuels archive.[12]

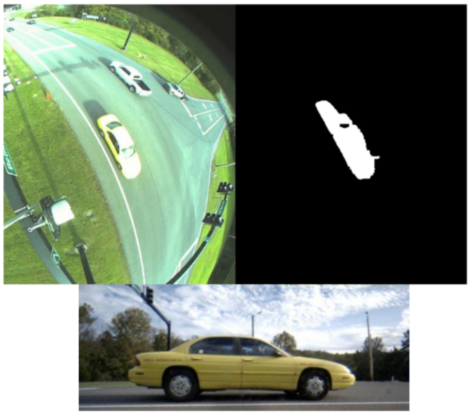

Figure 1. Examples of vehicle capture, segmentation mask, and ground-based sensor image from the OOVD data set.

Application of Data Set

The data set has multiple applications, including segmentation studies and shadow analysis. Here we demonstrate the utility of the set for estimating fuel and vehicle characteristics in three use cases (bounding box size, vehicle discrimination via classifiers that leverage bounding boxes, and a "fine grained" classification of vehicle make and model for a limited number of classes in the data set).

Fuel Consumption from Bounding Box

Our first experiment for determining fuel consumption estimates from overhead imagery used the vehicle bounding boxes generated by the aforementioned segmentation process. In this example we leveraged the typical locations of traffic in the intersection images, which we refer to as "NearLane", "TurnLane", and "FarLane". Our analysis focused on each of the three regions independently. For 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. Our results show that thresholds of 400 pixels for the NearLane, 350 pixels for the TurnLane, and 300 pixels for the FarLane separate high fuel consumers (average approximately 6 MPG) from the remaining low-to-moderate consumers; thus, 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, as shown in Figure 2.

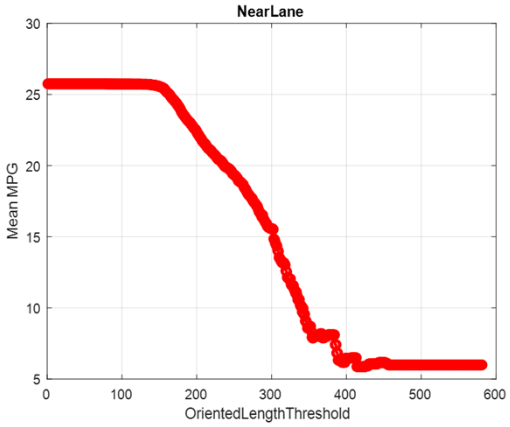

Figure 2. Average fuel consumption estimate in MPG for vehicles in the "near lane" based on the oriented bounding box length in OOVD. From left to right the average fuel consumption of vehicles with an oriented bounding box length greater than 0 has an average MPG estimate of approximately 26 MPG. As the size threshold increases, smaller vehicles are omitted, which tend to have better fuel economy in general. Vehicles above ~400 pixels have a mean MPG of approximately 6 MPG.

Vehicle Discrimination from Bounding Box

Our second analysis also used the bounding box but attempted to discriminate "regular" vehicles from the special vehicle classes of high fuel consumers in cases where the bounding box size overlaps between these broad classes. The goal here is to determine if more information regarding vehicle traffic can be obtained from the overhead camera imagery beyond relative vehicle size. To this end, we took the largest 100 "regular" vehicles from each and 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 Bus examples, we elected to combine this class with the 18Wheeler class. Three experiments were conducted on the NearLane, FarLane, and TurnLane. The overall data set was reduced to 60 random samples of each class (18WheelerBus, MultiAxle, DeliveryVan, Chevrolet Express Bus) and 60 random samples from the 100 largest examples of the "regular" vehicle class. A pre-trained convolutional neural network based on the MobilenetV2[13] topology was utilized to create a 1000-dimension feature vector using the output of the last fully connected layer. The data set of 300 vectors was split into five folds, with 210 vectors for training and 90 for testing. The classifier used an error-correcting output code model[14] for multiple classes and support vector machines. After each set of folds was completed, a new set was generated via random selection 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 1 lists example results for vehicles in the Near Lane.

Table 1. Confusion matrix for results in the near lane (accuracy 86%).

	18Wheeler+Bus	Chevrolet ExpressBus	Delivery Van	Multi-Axel	Other Vehicles
18Wheeler+Bus	45.8	0.50	7.2	5.5	1.0
Chevrolet ExpressBus	0.30	58.3	1.4	0.0	0.0
DeliveryVan	7.90	0.50	45.60	4.6	1.4
MultiAxel	4.50	1.00	3.90	49.7	0.90
Other Vehicles	0.30	0.80	0.90	0.90	57.1

Table 2. Accuracy and fuel consumption estimate for OOVD classes

CNN	Classifier Accuracy (at least 20 entries, 78 classes)	Fuel Consumption Estimate MPG	Classifier Accuracy (at least 40 entries, 29 classes)	Fuel Consumption Estimate MPG
ResNet50	28%	7.05	40%	4.83
ResNet101	27%	6.85	39%	4.75
GoogleNet	24%	7.27	37%	5.17
SqueezeNet	40%	5.81	52%	3.92
MobileNetV2	36%	5.94	44%	4.72
AlexNet	19%	8.04	32%	6.12
VGG-16	25%	6.68	34%	5.65
VGG-19	23%	7.44	34%	5.78

Fine Grained Discrimination

Our final experiments with OOVD explore the potential for fine-grained vehicle classification using solely overhead imagery. We performed two experiments, using the classes in the data set with at least 20 labeled vehicles and 40 labeled vehicles. We used the vehicle classes and randomly removed vehicle samples to create a balanced data set with at least 20 and 40 examples per set. (This process was repeated five times overall.) For each of these data sets, we performed four-fold validation testing. In the training phase, eight different pre-trained networks[15,16,17,18,19] were used from the MATLAB Deep Learning Toolbox[20], by extracting a feature vector from the output of the last fully connected layer. We again trained a classifier ensemble using an error-correcting output code model[14] for multiple classes with support vector machines. The average performance of the folds was utilized, and we estimated the fuel consumption error based on the vehicle MPG by assuming that if we successfully identified the make and model of the vehicle, our error was 0 MPG; otherwise, we used the MPG estimate from the erroneous classification to compute the MPG error. The results are shown in Table 2. For comparison, if the mean MPG estimate is used, the error would be 7.94 MPG and 8.24 MPG for the 20 example and 40 example cases. With this limited set of data, we achieve accuracy levels that are comparable to the results from larger studies as described in the next section. Also, the Squeezenet[15] and MobileNetV2 topologies have the best performance, which is likely because they have fewer parameters to train and therefore achieve better results with less data than the larger networks. (These also have the advantage of being more practical to deploy in an embedded system such as a traffic control device.)

Limitations of Visual Methods for Fuel Consumption

Because of our concerns regarding the ability to collect sufficient data from our field collection, we elected to explore the ability of fine-grained classifiers to estimate fuel consumption by using an existing data set[21]. The data set contains over 2500 fine-grained classes of vehicle make/model and year and also includes fuel economy estimates. We retrained a convolutional neural network based on the AlexNet topology[16] to act as a vehicle make/model classifier. This was inspired by the example of Gebru et al[21] and 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 overhead (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. The results are summarized in Table 3. Note that the error using the mean estimate is 6.1 MPG, thus the estimate tends to be less useful at 40 meters or more.

Table 3. Estimates of fuel consumption using baseline CNN model with Gebru data set

Range to vehicle (m)	Classifier Accuracy	RMS MPG Error
0	33%	3.5
20	16%	5.1
40	3%	6.7
60	1%	10.0

Application to Traffic Control

Our second focus was "teaching a grid of cameras to improve traffic mobility and fuel economy". Our simulation goal was to determine the feasibility of traffic control informed by overhead fisheye lens camera technology based on our data analysis in the preceding sections. We used reinforcement learning[22] (RL) to teach the controllers how to control the light timing. RL seeks to determine the best action to take given a sensed environment. RL networks typically accept a representation of the environment known as a "state." 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. 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. However, if there are longer time spans under consideration, such a reward may be too weak to enable successful learning, since the network would need to accurately predict the environmental state many time steps in the future.

There are multiple papers in the open literature concerned with applying RL to traffic control as well as energy usage[23,24]. In our approach we used a deep network[25] to generate traffic control changes, but we started with the assumption that the overhead camera includes "edge computing" to deliver a fuel consumption estimate. This approach starts with visual-compute models that perform vehicle segmentation and rudimentary classification. For our simulation platform, we selected the Simulation of Urban MObility (SUMO)[26] due to its history in transportation studies, and open-source availability. SUMO features a traffic control interface (TraCI) that allows control over how the simulation functions with respect to traffic conditions. We also integrated Keras/Tensorflow machine learning packages[27] for the RL algorithms.

We added a "visual sensor model" to the SUMO simulation environment based on our initial findings with the GRIDSMART camera and our estimates on fine-grained classification accuracy. In particular, we limited the sensing space to 60 meters around the intersection and used MPG RMS errors from the fine-grained estimates 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 meters apart. The traffic distribution was 50% buses/trucks and 50% passenger vehicles. Buses and trucks traveled north or south, whereas 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. Two traffic densities were used: "dense" simulations generate vehicles every 1–4 seconds until a total of 500 vehicles are utilized, and "sparse" simulations generated a vehicle once every 10 seconds. All results are averaged over 10 simulations.

Four different traffic control policies were tested: (1) 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; 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. For the RL policies, the state consists 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 the target light each second for the last 3 seconds; and for the second RL policy only, additional fuel usage estimates from the adjacent lights 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 incorporates 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.

Some key results of our preliminary simulations are shown in Figure 3 through Figure 5. The total fuel usage in gallons was provided by the SUMO output. Figure 3 shows that the visual policies have better performance for fuel consumption. There is 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 which is consistent with our findings using the oriented bounding box lengths. 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. The scatter plots in Figures 4 and 5 show more information about the learning process and resultant decisions. There are two distinct distributions; the left are passenger vehicles and the right are larger "gas-guzzling" busses and trucks. The control policy stops vehicles with no preference; the right distribution is especially 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 in Figure 5 show a reduction of the wait times for larger vehicles, which suggests the intersection is learning to allow them to pass without stopping, which saves energy.

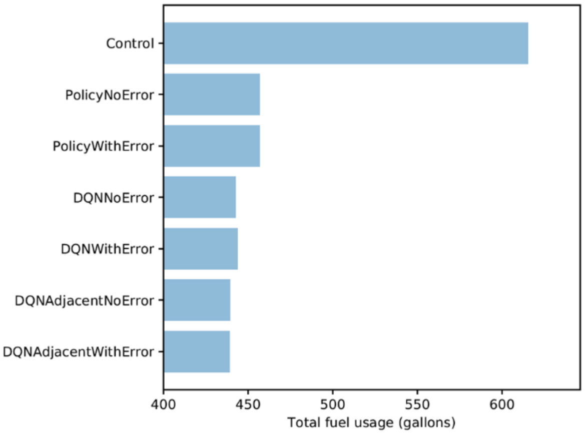

Figure 3 Fuel usage for different policies under the dense traffic experiments. The visual policies outperform the simple timing policy, with the RL methods performing slightly better.

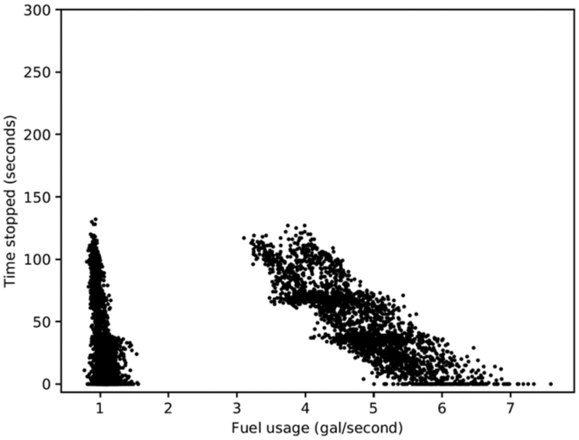

Figure 4 Control policy with dense traffic. The distribution on the left is the passenger vehicles whereas the distribution on the right is busses and trucks. The slant effect, particularly on the right, is caused by the decrease in the vehicle fuel consumption rate as vehicles are stopped for longer periods of time.

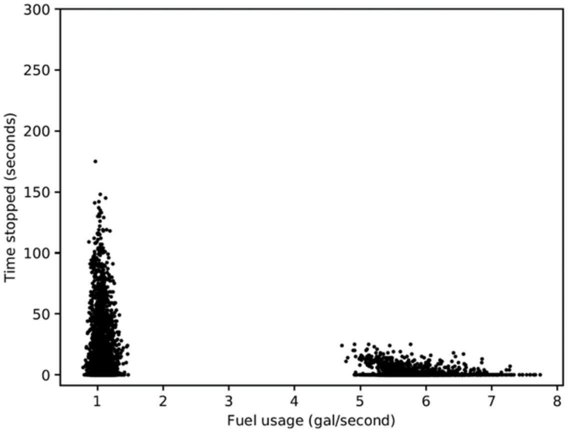

Figure 5 DeepQAdjacentWithError policy, under dense traffic. This model shows flattening of the right distribution indicating heavy fuel consumers stop for minimal times.

Conclusions

Improved transportation efficiency is vital to America's economic progress. This work was able to show the efficacy of building ground-truth data sets for vehicle classification from overhead traffic cameras that are currently used as sensors for traffic control. We introduced the OOVD data set for vehicle detection and classification. The utility of OOVD was shown by vehicle detection bounding box analysis, and two types of finer discrimination using CNNs. We demonstrated, through SUMO simulations that a deep-Q network with visual sensing could improve transportation efficiency. Sensing from adjacent intersections produced a better control policy, paving the way for future work on larger grids. 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 easily realized. Potential extensions of this work include field tests of the approach, additional simulations with larger grids and improved networks for both vehicle classification and traffic control. Future transportation systems may find the gains obtained by cameras with low detection resolution more difficult to realize, particularly as V2I self-identification becomes more common. Nevertheless, the role of sensing will be significant in intelligent transportation systems for some time to come.

Acknowledgments

We would like to acknowledge the support and assistance of Russ Henderson, Jonathan Sewell, Husain Aziz, Wael Elwasif, Thomas Naughton, John Turner, Jack Wells, Deborah Stevens, Kathy Jones, Rich Davies and Claus Daniel at ORNL.

This research used resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract DE-AC05-00OR22725. This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). The US government retains and the publisher, by accepting the article for publication, acknowledges that the US government retains a nonexclusive, paid-up, irrevocable, worldwide license to publish or reproduce the published form of this manuscript, or allow others to do so, for US government purposes. DOE will provide public access to these results of federally sponsored research in accordance with the DOE Public Access Plan (http://energy.gov/downloads/doe-public-access-plan). Work was funded by the Vehicle Technologies

Office HPC4Mobility seed project program of the US Department of Energy Office of Energy Efficiency and Renewable Energy.

References

[1] National Academy of Sciences, How We Use Energy, http://needtoknow.nas.edu/energy/energy-use/transportation/

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

[3] 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).

[4] Goodfellow, Ian, et al. 2016. *Deep Learning*. MIT Press, 2016.

[5] Yang, L., P. et al. 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.

[6] Dehghan, A., et al. 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.

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

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

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

[10] Bay, Herbert, et al. 2006. "Surf: Speeded up robust features." European conference on computer vision. Springer, Berlin, Heidelberg, 2006.

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

[12] EERE. 2019b. Alternate fuels data center site, https://afdc.energy.gov/data/10310.

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

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

[15] Forrest, N. Iandola, et al. "SqueezeNet: AlexNet-level accuracy with 50x fewer parameters and< 0.5 MB model size." *ICLR'17 conference proceedings*. 2017.

[16] Krizhevsky, Alex, et al. 2012. "Imagenet classification with deep convolutional neural networks." In Advances in neural information processing systems, pp. 1097–1105, 2012.

[17] He, Kaiming, et al. "Deep residual learning for image recognition." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016.

[18] Szegedy, Christian, et al. "Going deeper with convolutions." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015.

[19] Simonyan, Karen, and Andrew Zisserman. "Very deep convolutional networks for large-scale image recognition." *arXiv preprint arXiv:1409.1556* (2014).

[20] MathWorks. 2019b. Deep Learning Toolbox. https://www.mathworks.com/products/deep-learning.html

[21] Gebru, Timnit, et al. 2017. "Using Deep Learning and Google Street View to Estimate the Demographic Makeup of the US." arXiv preprint arXiv:1702.06683 (2017).

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

[23] 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.

[24] Al Islam, et al. 2018. "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.

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

[26] 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.

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

Author Biography

Thomas P Karnowski received his PhD from the University of Tennessee (2010). He has worked at Oak Ridge National Laboratory since 1990 on a variety of research projects in applications of signal and image processing.

Ryan Tokola received his MS in Electrical Engineering from the University of Michigan (2013). He has worked at Oak Ridge National Laboratory since 2014 on projects related to machine learning, computer vision, and image processing.

Sean Oesch received an MS in Software Engineering from Mercer University (2015). He worked at Robins Air Force Base designing software in support of the C-5 Galaxy Aircraft and has spent the last four years at Oak Ridge National Laboratory working on projects related to textual analytics and signal processing.

Matthew Eicholtz received a BS and MS in Mechanical Engineering from the Georgia Institute of Technology (2009, 2010) and a PhD in Mechanical Engineering from Carnegie Mellon University (2015). He was a postdoc at Oak Ridge National Laboratory from 2015-2018. He is currently an Assistant Professor of Computer Science at Florida Southern College.

Jeff Price earned a BS in Electrical Engineering from the U.S. Naval Academy and MS and PhD degrees from the Georgia Institute of Technology. Jeff worked as R&D Staff at Oak Ridge National Laboratory until 2008 when he joined GRIDSMART, eventually becoming their CTO. He is now VP of Technology after CUBIC's acquisition of GRIDSMART

Timothy F. Gee received his MS in Electrical Engineering from the Georgia Institute of Technology in 1993. He has primarily performed research and development in computer vision while at Oak Ridge National Laboratory for 15 years and then at GRIDSMART, A CUBIC Company for 11 years.