Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo, and M. Loper, eds.

MEASURING RELIABILITY OF OBJECT DETECTION ALGORITHMS FOR AUTOMATED DRIVING PERCEPTION TASKS

Huanzhong Xu

ICME
Stanford University
475 Via Ortega
Stanford, CA 94305, USA

Jose Blanchet

Department of Management Science and Engineering
Stanford University
475 Via Ortega
Stanford, CA 94305, USA

Marcos Paul Gerardo-Castro
Shreyasha Paudel

Green Field Labs
Ford Motor Company
3251 Hillview Ave
Palo Alto, CA 94304, USA

ABSTRACT

We build a data-driven methodology for the performance reliability and the improvement of sensor algorithms for automated driving perception tasks. The methodology takes as input three elements: I) one or various algorithms for object detection when the input is an image; II) a dataset of camera images that represents a sample from an environment, and III) a simple policy that serves as a proxy for a task such as driving assistance. We develop a statistical estimator, which combines I)-III) and a data augmentation technique, in order to rank the reliability of perception algorithms. Reliability is measured as the chance of collision given the speed of the ego vehicle and the distance to the closest object in range. We are able to compare algorithms in the (speed vs distance-to-closest-object) space using p-values and use this information to suggest improved-safety algorithms.

1 INTRODUCTION

The goal of this paper is to propose and investigate a comprehensive methodology for the performance reliability and the improvement of sensor algorithms for automated driving perception tasks.

Our goal is motivated by the need for having a scalable and fully data-driven method to ensure that self-driving vehicles can be safely deployed in public roads and highways. A self-driving system acting on erroneous or incomplete data is prone to making decisions that can lead to hazardous or catastrophic situations, jeopardizing the platform and surrounding safety. The quality of sensors and perception algorithms must be adequately assessed, as it conditions the system's ability to perceive the driving environment (Sivaraman and Trivedi 2013). To further robustify a self-driving system, it is also necessary to characterize the coverage of perception training data compared to the expected operational environment (Koopman and Wagner 2016). Safety evaluation systems for autonomous vehicles have received substantial attention in recent years to develop dependable perception systems (Levinson et al. 2011; Emzivat et al. 2018; Ramanagopal et al.

978-1-6654-3311-2/21/$31.00 ©2021 IEEE

2018). However, they do not offer performance guarantees which can be critical to the deployment of this technology.

As we shall discuss in Section 2 a significant portion of the literature on reliability focuses on the use of simulated scenarios to evaluate the performance of a class of algorithms which include both detection and identification. Instead, we seek to study a non-parametric data-driven approach. Another important difference between our methodology and those studied in the literature is that we not only evaluate the safety/reliability of an algorithm. Instead, our study also suggests ways to combine and improve the use of different algorithms in order to obtain more reliable estimators.

Our methodology takes as input three elements: I) one or various algorithms which detect, based on camera images, the distance from the vehicle to various objects; II) a set of data in the form of 2d camera images of a certain environment, and III) a simple policy which serves as a proxy for a driving task, such as driving assistance.

In the context of evaluating the reliability of a single algorithm, the methodology that we propose delivers a set of heat maps corresponding to the point estimates for the probability of collision as a function of the distance detected using the algorithm in question and the velocity at which the vehicle is traveling. We call this space, the (distance, velocity) space. We show that our non-parametric point estimates are consistent as we increase the sample size (see Theorem 1).

To estimate the probability of collision, we assume a constant deceleration, according to two levels of deceleration parameters corresponding to normal and emergency environments. The methodology also delivers corresponding asymptotically correct confidence regions for the probability of collision (see Theorem 2). We emphasize that our confidence intervals are derived with the optimal rate of convergence using non-parametric subsampling.

An immediate challenge that we face is that the set of annotated data is often not too large. This challenge is particularly important for the confidence regions. The dataset that we use is the KITTI dataset which contains 3768 images. So, we need to find a way to augment the dataset so that we can obtain reliable confidence intervals for the point estimates that we mentioned earlier. To overcome this problem, we introduce a statistical estimator which allows us to increase the dataset size at the expense of introducing the statistical assumption that the detected distance of various objects in a single image is conditionally independent given the image in question. By introducing this assumption, we increase the sample size by a factor of about 4, relative to the original dataset. This assumption, as we explain, does not seem to change the structure of the heat map, but it allows to mitigate the noise (i.e., statistical variance) compared to the original dataset size.

Monte Carlo simulation lies at the heart of the inference methodology that we use to produce confidence regions, based on the non-parametric theory of subsampling (see (Politis and Romano 1994)).

The most interesting use of the approach that we propose is, we believe, in the comparison of two or more algorithms, and the insights obtained by the possibility of combining them to improve reliability. In the context of evaluating the reliability of two algorithms, for example, our methodology delivers a map in the (distance, velocity) space which partition the areas in which both algorithms perform similarly or one of the algorithms performs better than the other one. These regions are coupled with the corresponding statistical p-values in a companion plot. Once again, these p-values are obtained using sub-sampling methods. The overall output suggests, for example, a way to combine algorithms with complementary skills in order to further improve reliability.

We showcase the use of our methodology in the study of three algorithms, in the context of monocular 3D object detection, corresponding to M3D-RPN (Brazil and Liu 2019), D4LCN (Ding et al. 2020), and Kinematic (Brazil et al. 2020). These algorithms were chosen as illustrations only. The data used, as we shall discuss, is annotated and obtained from the KITTI website (Geiger et al. 2012). Note that detection algorithms trained on KITTI dataset typically consider information including 3D locations, 2D bounding boxes, and classes of objects in images. In this paper, we will extract the distances of objects from their 3D bounding boxes and focus on them. Some of the insights found in our study are surprising relative to, for

instance, the performance indicators reported by KITTI. Of course, we emphasize that KITTI's indicators are different from those we consider, as they do not directly address the issue of safety as we model it here in connection with collisions. For instance, if an object is actually closer than what an algorithm perceives, is a more dangerous error in our setting than an error in estimation in the opposite direction (i.e. if the object is perceived to be closer than what it actually is). This asymmetry, it turns out, yields that some algorithms may perform better in the reliability sense that we consider compared to the pure prediction sense which is more traditional. We are not aware of any other work in the literature that makes this observation/distinction in the context of actual algorithms deployed in the literature. We believe that incorporating non-symmetric losses motivated by these types of different errors is a topic worthy of further scientific exploration.

The rest of the paper is organized as follows. In Section 2 we discuss related literature on the topic of reliability for self-driving tasks. We introduce our methodology in Section 3. In particular, the non-parametric estimator that we consider is given in Section 3.2 and confidence region methodology is given in Section 3.3. We then present how to compare and combine detection algorithms in Section 4. The experiment results and plots are shown in Section 5. Finally, we discuss final conclusions and future research directions motivated by our results in Section 6.

2 RELATED WORK

Naturalistic Field Operational Tests (N-FOT), which directly test automated vehicles under real-world driving conditions, is a popular way to test safety of automated vehicles. Unfortunately, this approach is challenging to implement because it is costly both in terms of time and resources to collect enough data in order to implement it. Moreover, only a relatively small proportion of daily driving data may include critical safety scenarios for testing reliability. Thus, a significant amount of literature focuses on the simulation of the environment and critical scenarios such as collision events via Monte Carlo methods – including acceleration and variance reduction techniques. For example, Zhao et al. (2015) introduced a new approach of Accelerated Evaluation (AE) which adopts Importance Sampling (IS) to accelerate the evaluation of automated vehicles safety. They introduce a parametric model of human-controlled vehicles derived from a real world dataset and use this parametric model to emphasize the critical scenarios and conduct statistical tests based on the parametric model. Then, IS is used to adjust the occurrence rate of critical scenarios back to the real world level. This method has been proved and applied to various tasks of interest, including lane change (Zhao et al. 2015), car-following maneuvers (Zhao et al. 2017), and frontal cut-in (Huang et al. 2017). The model of human-controlled vehicles can vary from single distribution models to piece-wise mixture distribution models (Huang et al. 2017), or be evaluated non-parametrically with Gaussian Mixture Models (Huang et al. 2018).

There are various differences between the approaches described earlier and our methodology. First, we do not evaluate the safety of a given policy. We are mostly interested in the safety of the sensor algorithms. We assume a fixed and simple policy, namely, the car decelerates at a given constant rate as a function of the velocity traveled and the distance to the nearest object detected in frontal view. Second, we use a non-parametric approach and deal with the problem of lack of data by introducing statistical assumptions. Finally, our method provides insights into how to improve sensor performance for reliability by combining algorithms.

3 GENERAL METHOD AND ESTIMATORS

3.1 Structure of the Pipeline

In this section we introduce a purely data-driven methodology for evaluating the reliability and safety of object detection algorithms while informed by simple autonomous driving tasks and environments. Generally, we try to estimate the probability that predictions of a detection algorithm may mislead the decisions made by the driving tasks under the given environment and lead to a collision event.

Our method evaluates performances of different detection algorithms in a given environment $\mathscr{E}$ sampled by a labeled dataset $\mathscr{D}$ of camera images (e.g., the KITTI dataset). For each image $\omega \in \mathscr{D}$, the dataset gives us the manually labeled locations ("true labels") of all objects, and an object detection algorithm $\mathscr{A}$ will try to predict these locations.

If a car is going to collide with an object, the most dangerous objects are those closest to it. For image $\omega \in \mathscr{D}$, let $D_{\min}(\omega)$ denote the distance of the closest object in ω according to the true labels. Let $\widehat{D}_{\min}(\omega)$ denote the prediction of $D_{\min}(\omega)$ by $\mathscr{A}$. Since the car relies on $\mathscr{A}$ to estimate "how close the car is to the object", the key distribution we want to focus on is $\mathbb{P}(D_{\min}|\widehat{D}_{\min})$. Ideally, we expect to have a joint distribution with $\mathbb{P}(D_{\min} = \widehat{D}_{\min}) = 1$.

Given the closest object's distance $D_{\min}$ and the car's velocity v, we can parametrize the event "whether a collision will happen" by $\text{Collision}(D_{\min}, v)$. The real-life modelling of $\text{Collision}(D_{\min}, v)$ should be detailed by the autonomous driving policy in use. For the sake of illustration, we will introduce a simple modelling of it in Section 3.2. With a given $\text{Collision}(\cdot,\cdot)$, we can estimate the probability of collision

$$\lambda(y,v) = \mathbb{P}\left(\text{Collision}(D_{\min}, v)\middle|\widehat{D}_{\min} = y\right),$$

i.e., how likely the car, under velocity v, is going to collide with an object which is predicted by $\mathscr{A}$ to be y meters away?

We also provide confidence levels to our estimations of function $\lambda(y,v)$. Generally, the more samples we have for the environment $\mathscr{E}$ (the larger $|\mathscr{D}|$ is), the more confident we are in our estimates. Hence the confidence levels also serve as an indicator whether we need to increase the size of $\mathscr{D}$ to improve the precision of $\lambda(y,v)$. The flowchart of our pipeline is shown in Figure 1.

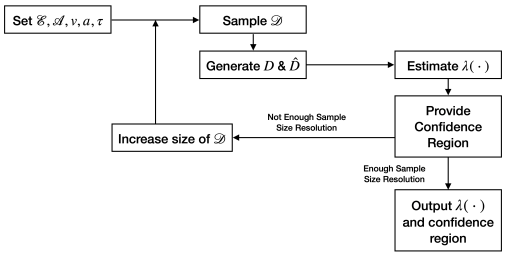

Figure 1: Flowchart of the pipeline.

3.2 A Consistent Non-parametric Estimator of $\lambda(y,v)$

To implement a simple model of "collision event", we consider a simple braking model – uniformly decelerating motion. Namely, whenever $\mathscr{A}$ gives us a small $\widehat{D}_{\min}$, we brake the car at a constant deceleration rate a. Suppose the reaction time the system needs to brake the car is t, we have

$$\text{Collision}(d,v) = \left\{ d < \frac{v^2}{2a} + tv \right\}.$$

Let $S(v;a,t) = \frac{v^2}{2a} + tv$ denote this "radius of collision" function parameterized by a and t, we have

$$\lambda(y,v) = \mathbb{P}\left(D_{\min} < S(v)\middle|\widehat{D}_{\min} = y\right).$$

We may get into trouble if we simply pair up $\widehat{D}_{\min}(\omega)$ and $D_{\min}(\omega)$ from each image $\omega \in \mathscr{D}$. The main concern here is that the detection algorithm $\mathscr{A}$ may not correctly locate the closest object for some images, as illustrated in Figure 2. Furthermore, this pairing procedure will only give us $|\mathscr{D}|$ pairs of data for estimation, which will limit our ability to improve the confidence level.

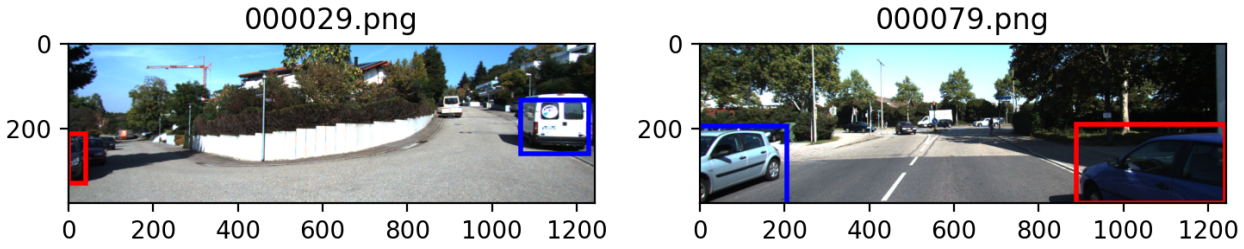

Figure 2: Example images where the closest object is different based on predictions (red boxes) and dataset labels (blue boxes).

To make sure $\mathbb{P}(D_{\min}|\widehat{D}_{\min})$ is estimated by data associated with same objects and fully explore the limits of our dataset, we need to consider including all objects in all images into our method. For each object labelled by the dataset, if we can find an object predicted by $\mathscr{A}$ which has the highest 2D and 3D Intersection over Union (IOU) overlap with it, then we pair up their distances. Such pre-processing will allow us to significantly increase the amount of data we can use, and make sure all pairs we will use are well-matched (i.e., referring to the same object in an image). The disadvantage is that we are considering objects not close to our car, so in order to estimate $\lambda(y,v)$ we will need some additional assumptions and extra computations detailed below.

For each image $\omega \in \mathscr{D}$, let $N(\omega)$ denote the number of labelled objects that we successfully pair up with a prediction. For the great majority of the images, this corresponds to the number of objects in the image. Let $n = \sum_{\omega\in\mathscr{D}} N(\omega)$ denote the total number of such objects in the dataset. By definition we have $D_{\min}(\omega) = \min_{1\leqslant i\leqslant N(\omega)} D_i(\omega)$ and $\widehat{D}_{\min}(\omega) = \min_{1\leqslant i\leqslant N(\omega)} \widehat{D}_i(\omega)$. Under certain assumptions and approximations, we are able to get a consistent estimator for $\lambda(y,v)$ and construct confidence intervals for it. We first explicitly give the estimator in Theorem 1, the proof of which is given in Appendix A, and then talk about construction of confidence of intervals in Section 3.3. Note that the consistency of the estimator paves the way for our estimation of confidence intervals, as we shall explain later.

Theorem 1 Assume $(D_i,\widehat{D}_i)_{i=1}^n$ are i.i.d. random vectors independent of N. Let $f_{\widehat{D}}$ denote the density of $\widehat{D}$, and $f_{\widehat{D}|D>z}$ denote the conditional density of $\widehat{D}$ given $D > z$ for some given z. Then a consistent estimator of $\widehat{\lambda}_n(y,v)$ as $n\to\infty$ is given by

$$1 - \frac{\widehat{f}_{\widehat{D}|D>S(v)}(y)\left(\frac{1}{n}\sum_{i=1}^n I\{D_i > S(v)\}\right)\sum_{m=0}^{\infty}\left(m\left(\frac{1}{n}\sum_{i=1}^n I\{D_i > S(v), \widehat{D}_i \geqslant y\}\right)^{m-1}\sum_{\omega\in\mathscr{D}} I\{N(\omega)=m\}\right)}{\widehat{f}_{\widehat{D}}(y)\sum_{m=0}^{\infty}\left(m\left(\frac{1}{n}\sum_{i=1}^n I\{\widehat{D}_i \geqslant y\}\right)^{m-1}\sum_{\omega\in\mathscr{D}} I\{N(\omega)=m\}\right)} \tag{1}$$

where $\widehat{f}_{\widehat{D}}$ and $\widehat{f}_{\widehat{D}|D>S(v)}$ are Gaussian kernel density estimations of $f_{\widehat{D}}$ and $f_{\widehat{D}|D>S(v)}$ respectively, and their bandwidths satisfy $\lim_{n\to\infty} h_n = 0$ and $\lim_{n\to\infty} nh_n = \infty$.

3.3 Subsampling and Point-wise Confidence Intervals

In this section we introduce Algorithm 1, which gives asymptotically valid point-wise confidence intervals for $\lambda(y,v)$. To construct asymptotically valid confidence intervals for $\lambda(y,v)$, Algorithm 1 uses the subsampling method by Politis and Romano (1994). We summarize its validity in Theorem 2 below, the proof of which is given in Appendix B. The only assumption necessary for applying the method is the following.

Assumption 1 Let $J_n(y,v)$ denote the sampling distribution of $\tau_n\left(\widehat{\lambda}_n(y,v) - \lambda(y,v)\right)$ for some non-decreasing normalizing constant τ_n. Then there exists a limiting law $J(y,v)$ with a continuous distribution such that $J_n(y,v)$ converges weakly to $J(y,v)$ as $n\to\infty$.

Since we can easily infer from the entire dataset $\mathscr{D}$ how many objects there are in any given image, but difficult to do so when we estimate $\lambda(y,v)$ based on a subset of $\left(D_i, \widehat{D}_i\right)$, the distribution of N will be treated as given in this section. We consider this to be a reasonable simplifying assumption since typically N will be supported on finitely many points and large deviations results will so learning the distribution of N is not complicated.

We now are ready to provide the statement of Theorem 2 and Algorithm 1.

Theorem 2 In the setting of Theorem 1, and assuming that $\widehat{D}$ has a density which is twice continuously differentiable, Algorithm 1 computes asymptotically correct point-wise confidence intervals for $\lambda(y,v)$.

Algorithm 1: Point-wise Confidence Intervals by Subsampling

def `lamb` (X_i, y, v)**:**
  $\widehat{f}_{\widehat{D}|D>S(v)}(y), \widehat{f}_{\widehat{D}}(y) \leftarrow$ `Gaussian_Kernel`$(X_i; h_b)$
  $\lambda'(y,v) \leftarrow$ By Equation (1)
  return $\lambda'(y,v)$
end
def `subsampling` (X, y, v)**:**
  $\widehat{\lambda}_n(y,v) \leftarrow$ By Equation (1)
  $b \leftarrow \sqrt{n}$ // Size of each batch
  $s \leftarrow \sqrt{n}$ // Number of batches
  $\tau_n, \tau_b \leftarrow n^{2/5}, b^{2/5}$
  for *i in* $1, 2, \cdots, s$ **do**
    $X_i \leftarrow$ Sample of size b without replacement from $\mathscr{D}$
    $\widehat{L}_i(y,v) \leftarrow \tau_b \left(\texttt{lamb}(X_i, y, v) - \widehat{\lambda}_n(y,v)\right)$
  end
  $\widehat{L}_i^s(y,v) \leftarrow \texttt{sort}\left(\widehat{L}_i(y,v)\right)$
  return $\left(\widehat{\lambda}_n(y,v) - \frac{1}{\tau_n}\widehat{L}^s_{0.95s}(y,v), \widehat{\lambda}_n(y,v) - \frac{1}{\tau_n}\widehat{L}^s_{0.05s}(y,v)\right)$
end

Remark 1 Another approach for building confidence intervals is based on Bootstrap. However, in the literature on non-parametric density estimation, generally the Bootstrap method is applied in conjunction with under-smoothing which deteriorates the rate of convergence, see Section 3.3.2 of Chen (2017). Instead, applying subsampling we maintain the optimal rate of convergence.

4 COMPARISON, COMBINATION AND RANK

We now use $\lambda(y,v)$ to compare the reliability of different detection algorithms and combine them to improve the overall performance. Let $\lambda^*(y,v)$ denote the reliability of an oracle detection model, i.e., a model that always predicts $\widehat{D}^* = D$. The interpretation of $\lambda^*(y,v)$ is that it estimates the probability of collision of our autonomous driving policy at different (y,v), assuming the vehicle is equipped with a perfect detection algorithm. Then naturally for $\lambda^{\mathscr{A}}$ measuring the performance of an arbitrary detection algorithm $\mathscr{A}$, the smaller $|\lambda^*(y,v) - \lambda^{\mathscr{A}}(y,v)|$ is, the more reliable $\mathscr{A}$ will be at (y,v). An important note here is that the sign of $\lambda^*(y,v) - \lambda^{\mathscr{A}}(y,v)$ is just as important as its absolute value. Namely, if $\lambda^*(y,v) < \lambda^{\mathscr{A}}(y,v)$, then $\mathscr{A}$ will mislead the driving policy by overestimating the danger, which may lead to unnecessary brakes and should be slightly penalized. If $\lambda^*(y,v) > \lambda^{\mathscr{A}}(y,v)$, then $\mathscr{A}$ will mislead the driving policy by underestimating the danger, which may lead to a collision event and should be heavily penalized. Thus, here we present a

weighted loss function $\ell(y,v)$ by

$$\ell(y,v;\lambda^*,\lambda^{\mathscr{A}}) = \begin{cases} \alpha(\lambda^{\mathscr{A}}(y,v) - \lambda^*(y,v)) \text{ if } \lambda^*(y,v) < \lambda^{\mathscr{A}}(y,v) \\ \beta(\lambda^*(y,v) - \lambda^{\mathscr{A}}(y,v)) \text{ if } \lambda^*(y,v) > \lambda^{\mathscr{A}}(y,v) \end{cases}$$

where $\alpha << \beta$, and the ratio between them is a user-defined tuning hyper-parameter for "how much more serious a collision event is compared with an overly cautious brake" (e.g. an error that would imply a collision is 10 times more important than an error that will not result in a collision.)

For two algorithms, we may directly use $\ell(y,v)$ to compare and combine their point-wise performances. Such comparison result is shown in Section 5.4, where we compare D4LCN and Kinematic. We will also illustrate how to use this result to determine which algorithm we should rely on at different (y,v).

For more than two algorithms, we introduce a score, which measures how much an individual detection model deviates from this oracle detection model on average, to compare and rank their performances. Specifically, let

$$L(\lambda^*,\lambda^{\mathscr{A}}) = \int \ell(y,v;\lambda^*,\lambda^{\mathscr{A}})h(y,v)\mathrm{d}(y,v), \tag{2}$$

i.e., the expectation of the loss $\ell(y,v)$ given the density function $h(y,v)$, which measures the relative importance of different (y,v). In Section 5.5 we will compute the scores for three detection algorithms and compare our ranking result with that on KITTI leaderboard.

5 RESULTS

5.1 Environment Setup

Throughout this section, we will use KITTI validation dataset as our labeled dataset $\mathscr{D}$, and compare the performances of three recently developed 3D detection models: M3D-RPN, D4LCN, and Kinematic. We set the reaction time $t = 0.1$s and test our estimator with two levels of deceleration: $a_{safe} = 3.92m/s^2$ and $a_{max} = 6.86m/s^2$. The function $\lambda(y,v;a,t)$ will be computed for all three detection algorithms and two deceleration levels. Section 5.2 will show the heat maps of $\lambda(y,v;a,t)$, and Section 5.3 will present example confidence intervals for these estimates. In Section 5.4 we will discuss how to compare and combine these detection algorithms to get an overall better estimator, and finally rank them in Section 5.5.

5.2 Heat Maps of $\lambda(y,v)$

Figure 3 shows the heat maps of $\lambda(y,v)$, where the rows correspond to the two levels of decelaration and the columns correspond to the three detection algorithms: M3D-RPN, D4LCN, and Kinematic respectively. In all plots, we also explicitly draw the black curve $y = S(v)$ to help understand the general pattern of $\lambda(y,v)$. The plots are similar, although there are significant differences which can already be observed. For example, note the irregularity in the blue area in Kinematic's heat map when $y > 50$ (relative to the other two algorithms). We will compare these heat maps numerically and explain the irregular (y,v)'s by introducing their p-values in following sections.

5.3 Point-wise Confidence Intervals

We will focus on deceleration level $a_{max} = 6.86m/s^2$ hereafter since the results of a_{safe} are quite similar. We take two frequently met speed limits in daily driving, $v = 25$ mph and $v = 40$ mph, as examples to show our confidence intervals for $\lambda(y,v)$. The results are shown in Figure 4.

We can easily see that our estimation of $\lambda(y,v)$ becomes highly unstable when $y \geqslant 50$. Lack of data is the main reason here, since almost 90% of objects in KITTI dataset are less than 50 meters away from the ego vehicle. As illustrated in Figure 1, we may need to collect more data in this range to improve our confidence of estimates.

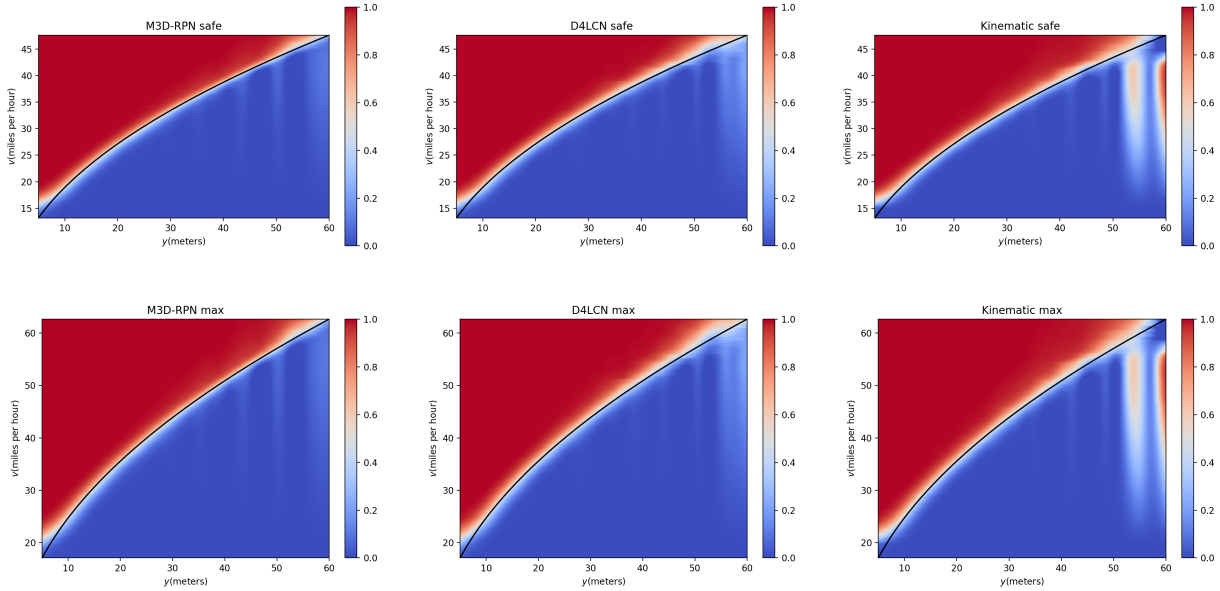

Figure 3: Heat maps of $\lambda(y,v)$'s for different detection algorithms and deceleration levels.

5.4 Comparison and Combination of Two Algorithms

We now compare D4LCN and Kinematic as an example and discuss how we can get an overall better estimator from them. As discussed in Section 4, we first set up an oracle detection algorithm and estimate λ^*, and then compare the differences between $\lambda^{\mathscr{A}}$ of the two detection algorithms with λ^* respectively. The result is shown in Figure 5. Note that as discussed in Section 5.3, we will not include the comparison when $y \geqslant 50$ because of lack of data.

For the green areas where both detection algorithms have their $\lambda^{\mathscr{A}}$ close enough to λ^*, we may choose either one to estimate the probability of collision. In other cases, we may choose to rely on D4LCN when (y,v) falls in an orange area and on Kinematic when (y,v) falls in a purple area.

We also present a plot of p-values for this comparison plot. Note that areas of different colors have different null hypotheses: the null hypothesis for orange area is $\ell(y,v;,\lambda^*,\lambda^D) > \ell(y,v;,\lambda^*,\lambda^K)$, for purple area is $\ell(y,v;,\lambda^*,\lambda^D) < \ell(y,v;,\lambda^*,\lambda^K)$, and for green area is $\max\{\ell(y,v;,\lambda^*,\lambda^D),\ell(y,v;,\lambda^*,\lambda^K)\} \geqslant 0.05$. Note that the uncertainty mostly happens around the transition from green to other areas or when $y \geqslant 50$. Thus, we are generally confident in the comparison result in orange and purple areas when $y < 50$.

5.5 Comparing Safety in Detection Algorithms

We compute the score introduced in Equation (2) for all three detection algorithms, with $h(y,v)$ being a uniform distribution. Note that then for any given $\lambda^{\mathscr{A}}$, L is a linear function of the ratio β/α. The slopes of this linear loss function are 0.0325, 0.0429, and 0.0694 for M3D-RPN, D4LCN, and Kinematic respectively. Therefore, we may conclude that M3D-RPN overall tends to provide safer estimates, and D4LCN yields safer estimates than Kinematic. This ordering is not aligned with that of the KITTI leader-board, which ranks Kinematic as the best detection algorithm. However, we emphasize that our criterion places more emphasis on overestimation of distances than on underestimation.

6 CONCLUSIONS AND FUTURE DIRECTIONS

In our analysis, we introduced simplifying assumptions (e.g. the distribution of N is known). The sub-sampling theory can be applied relaxing these assumptions. On the statistical side, we believe that the most

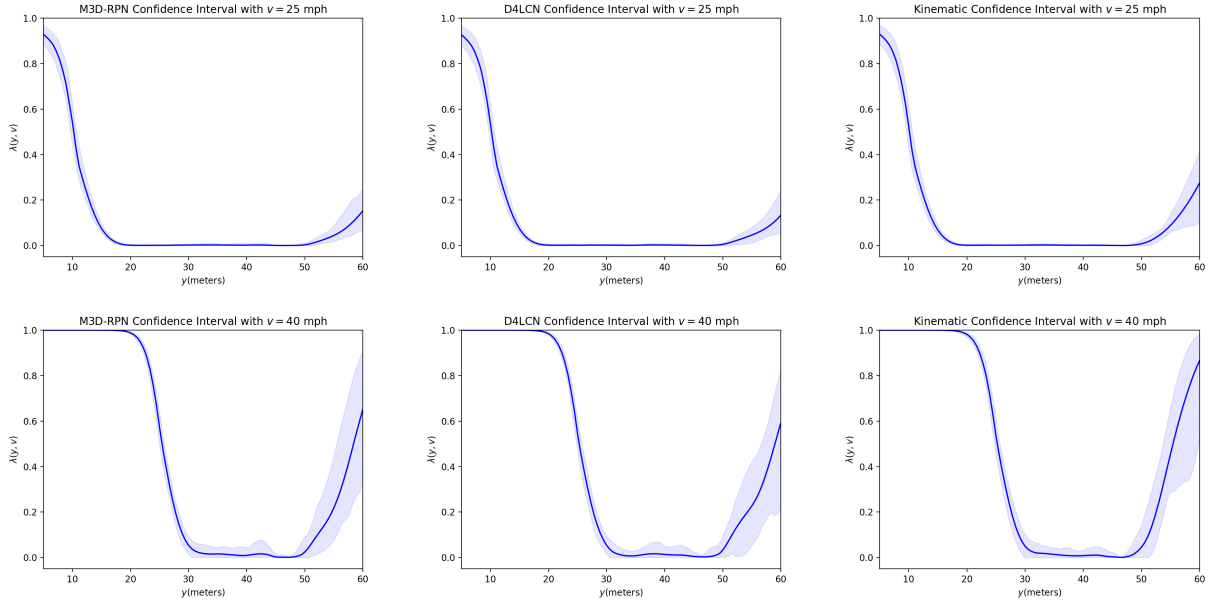

Figure 4: Point-wise confidence intervals of three detection algorithms at $v = 25$ and $v = 40$.

interesting issue is extending the non-parametric estimators to functional versions instead of point-wise estimation. Another topic of interest suggested by our research is the possibility of combining algorithms based on the context (e.g. velocity and speed) to improve overall reliability performance. Systematically studying this type of fusion is of significant interest. We also are interested in considering more complex data (e.g. videos). The challenge is that the data-augmentation techniques that we introduced (i.e. conditional independence of objects in images) are more difficult to apply. Finally, an additional direction involves considering more complex policies (beyond deceleration given a velocity field and the distance to the closes object) would be of significant interest as well.

ACKNOWLEDGMENTS

This project was made possible by funding from the Ford-Stanford Alliance. We thank Yueyang Liu for her support and development during the first part of this project. We thank Jinesh Jain and Viet-Anh Nguyen for their advice and assistance.

A Proof of Theorem 1

Proof. We first express the target function $\lambda(y,v) = \mathbb{P}(D_{\min} \leqslant S(v) | \widehat{D}_{\min} = y)$ in terms of $(D_i, \widehat{D}_i)_{i=1}^n$, and find consistent estimators for each component of the expression. We will write $\mathbb{P}(\widehat{D}_{\min} = y) = f_{\widehat{D}(y)} dy$, similarly with other density expressions. Observe that

$$\begin{aligned}
\mathbb{P}(D_{\min} \leqslant S(v) | \widehat{D}_{\min} = y) &= 1 - \mathbb{P}(D_{\min} > S(v) | \widehat{D}_{\min} = y) \\
&= 1 - \sum_{n=1}^{\infty} \mathbb{P}\left(D_{\min} > S(v) \middle| N = n, \widehat{D}_{\min} = y\right) \cdot \mathbb{P}\left(N = n \middle| \widehat{D}_{\min} = y\right) \\
&= 1 - \sum_{n=1}^{\infty} \mathbb{P}\left(D_{\min} > S(v) \middle| N = n, \widehat{D}_{\min} = y\right) \cdot \frac{\mathbb{P}(N = n)\mathbb{P}(\widehat{D}_{\min} = y | N = n)}{\mathbb{P}(\widehat{D}_{\min} = y)}.
\end{aligned}$$

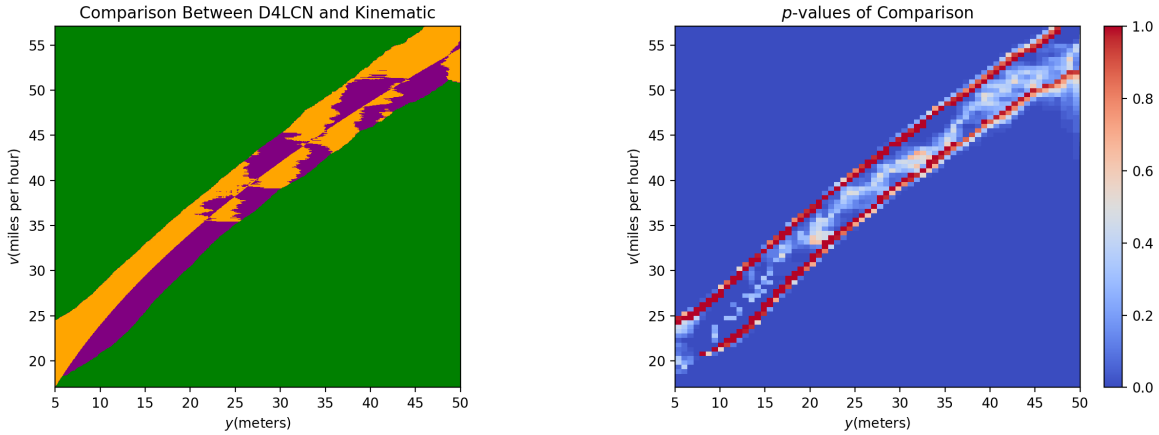

Figure 5: Orange areas correspond to (y,v)'s where D4LCN performs better (closer to the oracle detection algorithm), while purple areas correspond to those where Kinematic performs better. Green areas are where both algorithms have good performances (the differences between $\lambda^{\mathscr{A}}$ and λ^* are less than 0.05). We also include p-values which test the hypothesis implied by the color coding (e.g. for the orange area the null is that the Kinematic performs better, so rejecting with small p-values supports the claim that D4LCN performs better).

For the first term in the summation, we have

$$\begin{aligned}
&\mathbb{P}\left(D_{\min} > S(v) \middle| N=n, \widehat{D}_{\min}=y\right)\\
=&\sum_{i=1}^{n} \mathbb{P}\left(D_{\min} > S(v) \middle| \arg\min_{1\leqslant j\leqslant n} \widehat{D}_j = i, \widehat{D}_i = y\right) \cdot \mathbb{P}\left(\arg\min_{1\leqslant j\leqslant n} \widehat{D}_j = i, \widehat{D}_i = y \middle| N=n, \widehat{D}_{\min}=y\right)\\
=&\frac{1}{n}\sum_{i=1}^{n} \mathbb{P}\left(D_{\min} > S(v) \middle| \widehat{D}_1 \geqslant y, \ldots, \widehat{D}_{i-1} \geqslant y, \widehat{D}_i = y, \widehat{D}_{i+1} \geqslant y, \ldots, \widehat{D}_n \geqslant y\right)\\
=&\frac{1}{n}\sum_{i=1}^{n} \frac{\mathbb{P}(D_1 > S(v), \ldots, D_n > S(v), \widehat{D}_1 \geqslant y, \ldots, \widehat{D}_i = y, \ldots, \widehat{D}_n \geqslant y)}{\mathbb{P}(\widehat{D}_1 \geqslant y, \ldots, \widehat{D}_i = y, , \ldots, \widehat{D}_n \geqslant y)}\\
=&\frac{1}{n}\sum_{i=1}^{n} \mathbb{P}(D_i > S(v) | \widehat{D}_i = y) \prod_{j\neq i} \mathbb{P}(D_j > S(v) | \widehat{D}_j \geqslant y))\\
=&\frac{1}{n}\sum_{i=1}^{n} \mathbb{P}(D > S(v) | \widehat{D} = y) \left(\mathbb{P}(D > S(v) | \widehat{D} \geqslant y)\right)^{n-1} = \mathbb{P}(D > S(v) | \widehat{D} = y) \left(\mathbb{P}(D > S(v) | \widehat{D} \geqslant y)\right)^{n-1}.
\end{aligned}$$

For the second term in the summation, we have

$$\begin{aligned}
\mathbb{P}\left(\widehat{D}_{\min} = y \middle| N=n\right) &= \sum_{i=1}^{n} \mathbb{P}\left(\widehat{D}_1 \geqslant y, \ldots, \widehat{D}_i = y, \ldots, \widehat{D}_N \geqslant y\right)\\
&= \sum_{i=1}^{n} \mathbb{P}\left(\widehat{D} \geqslant y\right)^{n-1} \cdot f_{\widehat{D}}(y)dy = n f_{\widehat{D}}(y)dy \mathbb{P}\left(\widehat{D} \geqslant y\right)^{n-1},
\end{aligned}$$

and

$$\mathbb{P}(\widehat{D}_{\min} = y) = \sum_{m=1}^{\infty} \mathbb{P}\left(\widehat{D}_{\min} = y \middle| N=m\right) = \sum_{m=1}^{\infty} \mathbb{P}(N=m) \cdot m f_{\widehat{D}}(y)dy \left(\mathbb{P}(\widehat{D} \geqslant y)\right)^{m-1}.$$

Therefore we have

$$\begin{aligned}
\lambda(y,v) &= 1-\sum_{n=1}^{\infty}\mathbb{P}\left(D_{\min}>S(v)\middle|N=n,\widehat{D}_{\min}=y\right)\frac{\mathbb{P}(N=n)\mathbb{P}(\widehat{D}_{\min}=y|N=n)}{\mathbb{P}(\widehat{D}_{\min}=y)}\\
&= 1-\mathbb{P}(D>S(v)|\widehat{D}=y)\sum_{n=1}^{\infty}\left(\mathbb{P}(D>S(v)|\widehat{D}\geqslant y)\right)^{n-1}\frac{\mathbb{P}(N=n)\cdot nf_{\widehat{D}}(y)dy\left(\mathbb{P}(\widehat{D}\geqslant y)\right)^{n-1}}{\sum_{m=1}^{\infty}\mathbb{P}(N=m)\cdot mf_{\widehat{D}}(y)dy\left(\mathbb{P}(\widehat{D}\geqslant y)\right)^{m-1}}\\
&= 1-\mathbb{P}(D>S(v)|\widehat{D}=y)\frac{\mathbb{E}_N\left[N\left(\mathbb{P}(D>S(v),\widehat{D}\geqslant y)\right)^{N-1}\right]}{\mathbb{E}_N\left[N\left(\mathbb{P}(\widehat{D}\geqslant y)\right)^{N-1}\right]}\\
&= 1-\frac{f_{\widehat{D}|D>S(v)}(y)\mathbb{P}(D>S(v))}{f_{\widehat{D}}(y)}\cdot\frac{\mathbb{E}_N\left[N\left(\mathbb{P}(D>S(v),\widehat{D}\geqslant y)\right)^{N-1}\right]}{\mathbb{E}_N\left[N\left(\mathbb{P}(\widehat{D}\geqslant y)\right)^{N-1}\right]}.
\end{aligned}$$

By Law of Large Numbers, we know $\frac{1}{n}\sum_{i=1}^{n}I\{D_i>S(v)\}$, $\frac{1}{n}\sum_{i=1}^{n}I\{D_i>S(v),\widehat{D}_i\geqslant y\}$, $\frac{1}{n}\sum_{i=1}^{n}I\{\widehat{D}_i\geqslant y\}$, and $\sum_{\omega\in\mathscr{D}}I\{N(\omega)=m\}$ are consistent estimators of $\mathbb{P}(D>S(v))$, $\mathbb{P}(D>S(v),\widehat{D}\geqslant y)$, $\mathbb{P}(\widehat{D}\geqslant y)$, and $\mathbb{P}(N=m)$ respectively. As shown in Parzen (1962), a Guassian kernel density estimator with bandwidth h_n satisfying $\lim_{n\to\infty}h_n=0$ and $\lim_{n\to\infty}nh_n=\infty$ is a consistent estimator. Thus, $\widehat{f}_{\widehat{D}}$ and $\widehat{f}_{\widehat{D}|D>S(v)}$ are consistent estimations of $f_{\widehat{D}}$ and $f_{\widehat{D}|D>S(v)}$ respectively. Therefore by Slutsky's theorem, we finally conclude that

$$1-\frac{\widehat{f}_{\widehat{D}|D>S(v)}(y)\left(\frac{1}{n}\sum_{i=1}^{n}I\{D_i>S(v)\}\right)\sum_{m=0}^{\infty}m\left(\frac{1}{n}\sum_{i=1}^{n}I\{D_i>S(v),\widehat{D}_i\geqslant y\}\right)^{m-1}\sum_{\omega\in\mathscr{D}}I\{N(\omega)=m\}}{\widehat{f}_{\widehat{D}}(y)\sum_{m=0}^{\infty}m\left(\frac{1}{n}\sum_{i=1}^{n}I\{\widehat{D}_i\geqslant y\}\right)^{m-1}\sum_{\omega\in\mathscr{D}}I\{N(\omega)=m\}}$$

is a consistent estimator of $\lambda(y,v)$. □

B Proof of Theorem 2

Proof. By Theorem 2A of Parzen (1962) using a Gaussian kernel with bandwidth $h_n\in\Omega(n^{-1/5})$ for the kernel density estimator $\widehat{f}_{\widehat{D}|D>S(v)}(y)$, we know its convergence to $f_{\widehat{D}|D>S(v)}(y)$ is asymptotically normal with rate $n^{2/5}$. Similarly, with the same Gaussian kernel, $\widehat{f}_{\widehat{D}}(y)$ converges to $f_{\widehat{D}}(y)$ at the same rate $n^{-2/5}$. The asymptotic normal limit does not have mean zero, however, because the bias term assuming that the density is twice continuously differentiable is of order $O(n^{-2/5})$ and it is characterized explicitly (see section 3.3.3 of Chen (2017) and Scott (2015)). Nevertheless, this not a problem when applying the subsampling method because the existence of a continuous limit law is all what the method requires, according to Assumption 1. For the other three indicator estimators, Central Limit Theorem implies that they all converge to their expectations with rate $\sqrt{n}$. Since $\widehat{\lambda}_n(y,v)$ is a simple function of these component estimators which all converge to some normal distribution asymptotically, the delta method implies that its convergence is also asymptotically normal with rate $n^{-2/5}$. Since the normal distribution is continuous, we can apply Assumption 1 and the result follows. □

REFERENCES

Brazil, G., and X. Liu. 2019. "M3D-RPN: Monocular 3D Region Proposal Network for Object Detection". In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. October 27th-November 2nd, Seoul, Korea, 9287–9296.

Brazil, G., G. Pons-Moll, X. Liu, and B. Schiele. 2020. "Kinematic 3D Object Detection in Monocular Video". In *European Conference on Computer Vision*. August 23rd-28th, Virtual, 135–152.

Chen, Y.-C. 2017. "A Tutorial on Kernel Density Estimation and Recent Advances". *Biostatistics & Epidemiology* 1(1):161–187.

Ding, M., Y. Huo, H. Yi, Z. Wang, J. Shi, Z. Lu, and P. Luo. 2020. "Learning Depth-Guided Convolutions for Monocular 3D Object Detection". In *Conference on Computer Vision and Pattern Recognition*. June 14th-19th, Virtual, 1000–1001.

Emzivat, Y., J. Ibañez-Guzmán, H. Illy, P. Martinet, and O. H. Roux. 2018. "A Formal Approach for the Design of a Dependable Perception System for Autonomous Vehicles". In *International Conference on Intelligent Transportation Systems*. November 4th-7th, Maui, USA, 2452–2459.

Geiger, A., P. Lenz, and R. Urtasun. 2012. "Are We Ready for Autonomous Driving? The KITTI Vision Benchmark Suite". In *Conference on Computer Vision and Pattern Recognition*. June 16th-21st, Providence, USA, 3354–3361.

Huang, Z., Y. Guo, M. Arief, H. Lam, and D. Zhao. 2018. "A Versatile Approach to Evaluating and Testing Automated Vehicles Based on Kernel Methods". In *Annual American Control Conference (ACC)*. June 27th-29th, Milwaukee, USA, 4796–4802.

Huang, Z., H. Lam, D. J. LeBlanc, and D. Zhao. 2017. "Accelerated Evaluation of Automated Vehicles Using Piecewise Mixture Models". *IEEE Transactions on Intelligent Transportation Systems* 19(9):2845–2855.

Huang, Z., D. Zhao, H. Lam, D. J. LeBlanc, and H. Peng. 2017. "Evaluation of Automated Vehicles in the Frontal Cut-in Scenario – An Enhanced Approach Using Piecewise Mixture Models". In *IEEE International Conference on Robotics and Automation*. May 29th-June 3rd, Marina Bay Sands, Singapore, 197–202.

Koopman, P., and M. Wagner. 2016. "Challenges in Autonomous Vehicle Testing and Validation". *SAE International Journal of Transportation Safety* 4(1):15–24.

Levinson, J., J. Askeland, J. Becker, J. Dolson, D. Held, S. Kammel, J. Z. Kolter, D. Langer, O. Pink, V. Pratt et al. 2011. "Towards Fully Autonomous Driving: Systems and Algorithms". In *IEEE Intelligent Vehicles Symposium*. June 5th-9th, Baden-Baden, Germany, 163–168.

Parzen, E. 1962. "On Estimation of a Probability Density Function and Mode". *The annals of mathematical statistics* 33(3):1065–1076.

Politis, D. N., and J. P. Romano. 1994. "Large Sample Confidence Regions Based on Subsamples under Minimal Assumptions". *The Annals of Statistics* 22(4):2031–2050.

Ramanagopal, M. S., C. Anderson, R. Vasudevan, and M. Johnson-Roberson. 2018. "Failing to Learn: Autonomously Identifying Perception Failures for Self-driving Cars". *IEEE Robotics and Automation Letters* 3(4):3860–3867.

Scott, D. W. 2015. *Multivariate Density Estimation: Theory, Practice, and Visualization*. John Wiley & Sons.

Sivaraman, S., and M. M. Trivedi. 2013. "Looking at Vehicles on the Road: A Survey of Vision-based Vehicle Detection, Tracking, and Behavior Analysis". *IEEE transactions on intelligent transportation systems* 14(4):1773–1795.

Zhao, D., X. Huang, H. Peng, H. Lam, and D. J. LeBlanc. 2017. "Accelerated Evaluation of Automated Vehicles in Car-following Maneuvers". *IEEE Transactions on Intelligent Transportation Systems* 19(3):733–744.

Zhao, D., H. Peng, H. Lam, S. Bao, K. Nobukawa, D. J. LeBlanc, and C. S. Pan. 2015. "Accelerated Evaluation of Automated Vehicles in Lane Change Scenarios". In *Dynamic Systems and Control Conference*. October 28th-30th, Columbus, USA, V001T17A002.

AUTHOR BIOGRAPHIES

HUANZHONG XU is a Ph.D. student at the Institute of Computational and Mathematical Engineering at Stanford University. His research interests lie in the areas of Statistics, Artificial Intelligence and Computational Methods. His email address is xuhuanvc@stanford.edu.

JOSE BLANCHET is a Professor of MS&E at Stanford University, from which he earned his doctorate degree. He serves on the editorial board of various journals in these areas, including Mathematics of Operations Research and Stochastic Systems, among others. His email address is jose.blanchet@stanford.edu. His website is https://web.stanford.edu/ jblanche/.

MARCOS PAUL GERARDO-CASTRO is a Research Scientist at Greenfield Labs, Ford Motor Company. He earned his Ph.D. in Robotics from the Australian Centre of Field Robotics at University of Sydney. His work focuses on sensor fusion of multiple sensing modalities, perception and machine learning. His email address is mgerard8@ford.com.

SHREYASHA PAUDEL is a Research Engineer at Greenfield Labs, Ford Motor Company. She earned her Masters in Aeronautics and Astronautics from Stanford University. She has research experience in sensor fusion and robotics, and is currently interested in studying uncertainty and bias in artificial intelligence algorithms. Her email address is: spaudel@ford.com.