

# Road Boundary Estimation Using Sparse Automotive Radar Inputs

Aaron Kingery and Dezhen Song

**Abstract**—Low-cost millimeter wavelength automotive radar can work effectively under low visibility or low reflection conditions caused by lighting, weather, pollution, or object surface properties when a camera or a lidar may fail. It can serve as a fallback solution to improve safety in autonomous driving. However, after filtering, radar signals tend to be sparse and noisy which poses new challenges in scene understanding. This paper presents a new approach to detecting road boundaries based on sparse radar signals. We model the roadway using a homogeneous model and derive its conditional predictive model under known radar motion. Using this predictive model and modeling radar points using a Dirichlet Process Mixture Model, we employ Mean Field Variational Inference (MFVI) to derive an unconditional road boundary model distribution. To generate initial candidate solutions for the MFVI, we develop a custom Random Sample and Consensus (RANSAC) variant to propose unseen model instances as candidate road boundaries. For each radar point cloud we alternate the MFVI and RANSAC proposal steps until convergence to generate the best estimate of all candidate models. We select the candidate model with the minimum lateral distance to the radar on each side as the estimates of the left and right boundaries. We have implemented the proposed algorithm and it has shown satisfactory results. More specifically, the mean lane boundary estimation error is not more than 11.0 cm.

## I. INTRODUCTION

Low-cost automotive radars can provide reliable sensory input under low visibility or low reflection conditions caused by lighting, weather, air pollution, or object surface properties when a camera or lidar may fail. If we can develop effective scene understanding techniques based on radar data, it will provide a fallback solution to enhance the safety of autonomous vehicles. Here we report a new road boundary detection algorithm based on radar data.

Under radar data, a road boundary refers to the boundary between different types of terrain or separations based on geometry. For example, asphalt pavement and grassy road shoulders have different radar reflective patterns. The boundary between them is the road boundary. Traffic cones, road curbs, or guard rails can also form a road boundary. It is clear that road boundaries are not necessarily the same as lane boundaries. Detection of road boundaries is important to ensure vehicle safety. Radar signals are typically relegated only to the detection of dynamic objects such as other vehicles, bicycles, and pedestrians because they are sparse and noisy, which presents a challenge for road boundary

A. Kingery and D. Song are with CSE Department, Texas A&M University, College Station, TX 77843, USA. D. Song is also with the Robotics Department of Mohamed Bin Zayed University of Artificial Intelligence (MBZUAI) in Abu Dhabi, UAE. Email: dezhen.song@mbzuai.ac.ae.

This work was supported in part by General Motors/SAE Autodrive Challenge II and by MBZUAI.

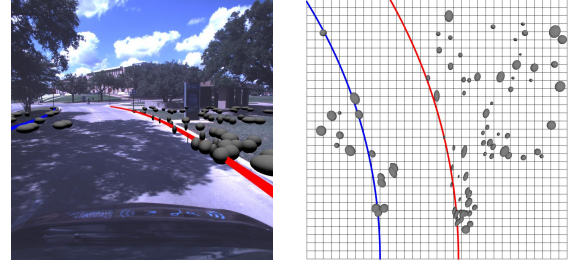


Fig. 1. An example of the radar road boundary estimation problem. On the left, the scenario is shown from the camera perspective, with results from the radar frame projected onto the camera image. On the right, a top down orthographic perspective of the scene. Grey ellipses are radar target detections, and the blue and red curves are the left and right road boundaries, estimated from the radar, respectively. The grid has 1 meter spacing.

detection. Unlike the significant existing development in lane detection in the vision or lidar domain, detection of the road boundary based on the sparse radar input is a new problem.

We propose a new method for detecting road boundaries based on sparse radar signals. By sparse signals, we refer to the filtered output of common automotive radars rather than the raw reflectivity images that are only available to radar developers. Fig. 1 illustrates the radar inputs and our problem. We model the roadway using a homogeneous four-coefficient arc model and derive its conditional predictive form under known radar motion. Using the conditional predictive model and model radar points using a Dirichlet Process Mixture Model (DPMM), we employ Mean Field Variational Inference (MFVI) to derive an unconditional road boundary model distribution. To generate initial candidate solutions for the MFVI, we develop a custom Random Sample and Consensus (RANSAC) variant to propose unseen model instances as candidate road boundaries. For each radar point cloud we alternate the MFVI and RANSAC proposal steps until convergence to generate the best estimate of all candidate models. We select the candidate model with the minimum lateral distance to the radar on each side as the estimates of the left and right boundaries.

We have implemented the proposed algorithm in C++ as a module in the Robot Operating System (ROS). We tested the algorithm on data collected from a Continental ARS430 automotive Doppler radar and it has shown satisfactory results. More specifically, the mean lane boundary estimation error is not greater than 11.0 cm, which is reasonable when considering the radar wavelength.

## II. RELATED WORKS

Detection of lane boundaries is a well-studied problem in the vision and lidar domains, but is new for road boundaries

under sparse radar signals [1]. Vision and lidar approaches have been the popular sensor choices, including camera-radar fusion [2] [3] and lidar-radar fusion [4], of which [5] provides a survey. However, vision and lidar are easily affected by severe weather conditions or other environmental factors, leading to poor performance. On the other hand, radars are less sensitive to environmental conditions. In general, when a radar is used in robot/vehicle perception, there are, broadly, two forms of output to consider: radar reflectivity images and radar target detection point clouds.

Radar reflectivity images are analogous to a camera image but with cells instead of pixels and with cell dimensions typically including at least range and azimuth. However, the radar reflectivity image is often much less sharp and has lower resolution than a camera image or a lidar point cloud. When using radar reflectivity images for the estimation of road boundaries, Nikolova and Hero [6] model the boundaries in the polar coordinate space of the radar reflectivity image and identify them as the edges of continuous homogeneous regions in the image with constant width and curvature. Kaliyaperumal et al. [7] propose a deformable model for the boundaries, a likelihood function to match the model with the edges in the image, and use the Metropolis-Hastings algorithm with simulated annealing to find the optimal match. Guo et al. [8] present the stripe Hough transform which is capable of detecting lines in the reflectivity image when there are orthogonal deviations from the boundary. Werber et al. [9] estimate straight line landmarks from the reflectivity image and associate them over time for vehicle localization.

For most commercially available automotive radars, like the one used for this work, the reflectivity image is not an available output. Instead, the image is downsampled into a point cloud of target detections generally representing points of peak reflectivity in the image. This radar point cloud data, while similar in format to lidar data, is comparatively sparse, noisy, and unstructured. As a consequence, lidar-based boundary detection approaches are generally not applicable to radar point clouds. A common technique to overcome these issues for radar point clouds is to temporally ‘stack’ the point clouds by transforming the points into a static coordinate frame using some known localization (e.g. GPS), merging them, and then estimating using this more dense stacked point cloud. Lundquist et al. [10] present two such approaches, estimation from an occupancy grid map and estimation from a Quadratic Program (QP) with prior sorting of target detections into left and right sets and outlier rejection. Xu et al. [11] generate an occupancy grid from temporally stacked radar observations and combine edge detection and RANSAC in order to identify linear boundaries.

A significant weakness of the stacking based approach is that it relies upon robust and accurate localization over the stacking window. When the localization is not accurate, especially when traversing turns and corners, the stacked radar point cloud often exhibits a distinct ‘smearing’ of the points which can degrade or destroy the estimation quality. For this reason, the algorithm we propose tracks the road

boundaries over time, applying new observations to update the existing estimate in a Bayesian manner. Among such approaches, Lee et al. [12] provide an instantaneous estimate of the road curvature and Lundquist et al. [10] track points and quadratic curve segments along the roadway as extended objects via an Extended Kalman Filter (EKF), but neither approach estimates the road boundaries. Lundquist et al. [13], as a prior step towards radar road intensity mapping, perform a K-means regression clustering of cubic curves in highway scenarios. Compared with this work, our algorithm directly estimates the primary road boundaries themselves, allows a dynamic number of candidate tracks, and makes a probabilistic, as opposed to binary, assignment of target detections to tracks, giving the algorithm robustness in the presence of the obfuscating clutter common in non-highway scenarios.

### III. PROBLEM DESCRIPTION

Consider that we have a vehicle equipped with automotive radar traveling along a roadway and we would like to estimate the location of the left and right road boundaries relative to the sensor at the time  $t$  of each measurement. Let us define  $\{\mathcal{D}_t\}$  as the dynamic radar ego coordinate system at time  $t$  that moves with the radar so that the  $X$ -axis is forward and the  $Y$ -axis is to the right.

#### A. Assumptions

- a.0 There exist road boundaries within the radar field of view.
- a.1 The radar is mounted such that its forward direction is parallel with the vehicle’s longitudinal axis.
- a.2 The radar is positioned such that it is located between the left and right road boundaries.
- a.3 The primary left and right road boundaries within the field of view (i.e. ignoring discontinuities at intersections, driveways, etc.) can be approximated as a circular arc or line in the Cartesian coordinate space.
- a.4 The motion of the radar between two subsequent time steps is known or estimated. For this purpose, we use the instantaneous radar ego-velocity estimation [14], however, other odometry modalities (e.g. inertial) should also be sufficient.

It should be noted that Assumption a.0 requires the presence of road boundaries to satisfy the detection condition. Although a road boundary does not exist in large open space under the same terrain (e.g. when a vehicle is crossing a large intersection) and it would limit the application scope of this algorithm by excluding intersections and very wide open roads, it is still a good fallback solution to ensure vehicle safety under severe weather conditions, because it can effectively protect the vehicle from veering off the road or colliding with roadside objects. According to a Virginia Department of Motor Vehicles report, a vehicle veering off the road is a factor in almost half of all fatal crashes in Virginia [15].

## B. Sensor Model

We will follow the sensor model in our previous work with these radars [14]. For completeness, we reiterate it here. Our sensor is a 77GHz automotive Doppler radar which periodically transmits a pulse and reports a set of the received reflections as estimated target detections. We use  $\mathbf{Z}_t = \{\mathbf{z}_{t,1}, \dots, \mathbf{z}_{t,n}\}$  to denote the set of target detections received as a datagram from a transmitted radar pulse. We consider each detection  $\mathbf{z}_{t,i}$  as a noisy observation of some ground truth measurement source  $\mathbf{s}_{t,i}$ . Each target detection is reported in polar coordinates in the radar coordinate frame  $\{\mathcal{D}_t\}$  and consists of the measurements

$$\mathbf{z}_{t,i} = [r_{t,i} \quad \theta_{t,i}]^\top = \mathbf{s}_{t,i} + \mathbf{v}_{t,i}, \quad (1)$$

where,  $i \in \{1, \dots, N\}$  indicates the target index,  $r_{t,i} \in [0, r_{\max}]$  is the  $i$ -th target range,  $\theta_{t,i} \in [\theta_{\min}, \theta_{\max}]$  is the  $i$ -th target azimuth where  $\theta_{t,i} = 0$  indicates the forward direction,  $\theta_{t,i} < 0$  indicates a target to the left, and  $\theta_{t,i} > 0$  indicates a target to the right, and  $\mathbf{v}_{t,i}$  is the observation noise such that  $\mathbf{v}_{t,i} \sim \text{Normal}(\mathbf{0}, \Sigma_{t,i})$  where  $\Sigma_{t,i} = \begin{bmatrix} \sigma_{r,i}^2 & 0 \\ 0 & \sigma_{\theta,i}^2 \end{bmatrix}$ . The Cartesian parameterization of the target position in  $\{\mathcal{D}_t\}$  is defined in the usual manner,  $[x_{t,i} \quad y_{t,i}]^\top = [r_{t,i} \cos(\theta_{t,i}) \quad r_{t,i} \sin(\theta_{t,i})]^\top$ .

## C. Road Boundary Model

In the Cartesian coordinate space, roadways are generally constructed as a series of circular arcs connected by linear segments, which may be considered as circular arcs with infinite radius. We desire to use a single model which can represent both, and, for this reason, we make use of the quadratic representation of a conic with constant curvature,

$$\beta_1(x^2 + y^2) + \beta_2x + \beta_3y + \beta_4 = 0, \quad (2)$$

where  $\beta_1 \neq 0$  represents a circle and  $\beta_1 = 0$  a line.

As the radar makes observations in polar coordinates, it is desirable to express the model in polar coordinates as well so we have,

$$\beta_1 r^2 + \beta_2 r \cos(\theta) + \beta_3 r \sin(\theta) + \beta_4 = 0. \quad (3)$$

From this we define the road boundary as the set of points

$$\{\mathbf{s} \mid \beta^\top \phi(\mathbf{s}) = 0, \mathbf{s} \in [0, r_{\max}] \times [\theta_{\min}, \theta_{\max}]\} \quad (4)$$

where  $\mathbf{s} = [r \quad \theta]^\top$  is a point expressed in polar coordinates,  $\phi(\mathbf{s}) = [r^2 \quad r \cos(\theta) \quad r \sin(\theta) \quad 1]^\top$  are the model basis functions, and  $\beta = [\beta_1 \quad \beta_2 \quad \beta_3 \quad \beta_4]^\top$  are the model coefficients. We note that the model is homogeneous, i.e.  $\beta \equiv c\beta$ , for all  $c \in \mathbb{R}$ .

## D. Problem Definition

Given a series of radar datagrams  $\mathbf{Z}_0, \dots, \mathbf{Z}_t$ , we will estimate the left and right road boundaries at each time  $t$ , with  $\beta_{t,l}$  and  $\beta_{t,r}$  as the coefficients of our road boundary model.

## IV. ALGORITHM

Our algorithm operates in three primary phases: model prediction from known radar motion, model estimation alternating candidate updates from the radar observations and proposing new candidates, and algorithm termination where we output the boundary estimates and prepare for subsequent radar observations. We provide a graphical overview of the flow of our algorithm in Fig. 2.

### A. The Roadway as a Dirichlet Process Mixture Model

As described in III-B, we model each target detection  $\mathbf{z}_{t,i}$  as the detection of some true source point  $\mathbf{s}_{t,i}$  with zero-mean Gaussian observation noise  $\mathbf{v}_{t,i}$ . We model the radar points as being generated by a DPMM, where the points are classified into two groups: outliers whose source is not necessarily modeled and inliers which are generated by a candidate model. Therefore, we will have  $K_t + 1$  models where model index  $k = 0$  indicates the outlier model and  $k \in \{1, \dots, K_t\}$  indicates a candidate model. Each observed target  $\mathbf{z}_{t,i}$  is associated with the latent variable  $c_{t,i} \sim \text{Categorical}(\pi_t)$  representing which model produced the target detection, where  $\pi_t = [\pi_{t,0} \quad \dots \quad \pi_{t,K_t}]^\top \sim \text{Dirichlet}(\alpha_t)$  are the mixture weights and  $\alpha_t = [\alpha_{t,0} \quad \dots \quad \alpha_{t,K_t}]^\top$  is the prior mixture concentration.

Ideally, one would represent the source point as its own latent variable described by a Spatial Distribution Model (SDM) on the source object. This is difficult in practice as such an SDM is heavily dependent upon the geometry of the scene and would need to account for occlusions, object shape, etc. Additionally, estimating the source point as a latent variable significantly complicates posterior inference unless the SDM is in the exponential family. Instead, we rely upon a Greedy Association Model (GAM) as in [16], and we will show that this produces a very straightforward algorithm for inference. The result is that we convert  $\mathbf{z}_{t,i}$  into a pseudo-observation of the model  $\beta_{t,k}$  via the implicit shape function of our model  $h(\mathbf{z}_{t,i}, \beta_{t,k}) = \beta_{t,k}^\top \phi(\mathbf{z}_{t,i})$ , as in [17]. Propagating the measurement uncertainty, we can say that  $h(\mathbf{z}_{t,i}, \beta_{t,k}) \sim \text{Normal}(0, \sigma_{ik}^2)$  where  $\sigma_{ik}^2 = \beta_{t,k}^\top \Phi(\mathbf{z}_{t,i}) \Sigma_{t,i} \Phi(\mathbf{z}_{t,i})^\top \beta_{t,k}$  with  $\Phi(\mathbf{z}_{t,i})$  being the Jacobian of  $\phi(\mathbf{z}_{t,i})$ . We note that this GAM results in a biased estimation for curved shapes for which [16] identifies and prescribes a correction. However, in the case of road boundary estimation with radar, the measurement uncertainty is not typically large enough relative to the curvature for this bias to have a significant effect. Thus, we consider  $E[h(\mathbf{z}_{t,i}, \beta_{t,k})] = 0$  to be a reasonable approximation. Finally, we model the outlier targets simply as observations from a uniform SDM over the field of view,  $\mathbf{z}_{t,i} \mid c_{t,i} = 0 \sim \text{Uniform}([0, r_{\max}] \times [\theta_{\min}, \theta_{\max}])$ .

Due to the fact that our road boundary model is homogeneous, it is necessary to choose a scale for the coefficients. Setting the scale such that  $\beta_{t,k}^\top \beta_{t,k} = 1$  is a natural choice. Consequently, the coefficients of each candidate model are distributed such that  $\beta_{t,k} \sim \text{Bingham}(\mathbf{C}_{t,k}^{-1})$ , which is the analogue to the normal distribution but conditioned on the

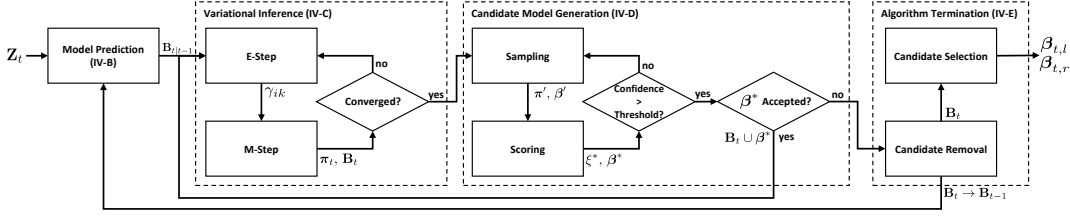


Fig. 2. Algorithm Flow Chart.

unit hypersphere [18]. The Bingham distribution has found related use in Quaternion Bingham Filters e.g. [19]–[21], which provide more detail on the properties of the Bingham distribution. Note that the Bingham distribution is typically parameterized by the eigen decomposition  $\mathbf{C}_{t,k}^{-1} = \mathbf{V}\mathbf{\Lambda}\mathbf{V}^T$ , however, for our purposes it will be more natural to work with  $\mathbf{C}_{t,k}^{-1}$  directly. While ideally it should be the case that  $p(\beta_{t,k}) = p(c\beta_k)$  where  $c \in \mathbb{R}$ , for our distribution, it is only true that  $p(\beta_{t,k}) = p(-\beta_{t,k})$ , i.e. the distribution is antipodally symmetric, but it will serve as a reasonable approximation in return for useful mathematical properties; namely that the Bingham distribution is in the exponential family, making the inference more tractable. Finally, it is the case that  $E[\beta_{t,k}] = \mathbf{0}$ , however, this trivial solution will not be useful for inference. Therefore, we will instead use the mode of the distribution which is the eigenvector corresponding to the minimum eigenvalue of the eigenvalue problem  $\mathbf{C}_{t,k}^{-1}\beta_{t,k} = \lambda\beta_{t,k}$ . Thus, with some abuse of notation, we will say that  $E[\beta_{t,k}]$  is equal to this mode.

In summary, we consider the generative model of radar detections to be

- $N_t$ : Number of detections.
- $K_t$ : Number of candidate models.
- $\alpha_t$ : prior concentration.
- $\pi_t \sim \text{Dirichlet}(\alpha_t)$ .
- $c_{t,i} \sim \text{Categorical}(\pi_t)$ .
- $\mathbf{z}_{t,i} \sim \text{Uniform}([0, r_{\max}] \times [\theta_{\min}, \theta_{\max}])$  if  $c_{t,i} = 0$ .
- $\beta_{t,k}^T \phi(\mathbf{z}_{t,i}) \sim \text{Normal}(0, \sigma_{ik}^2)$  if  $c_{t,i} = \{1, \dots, K_t\}$ .
- $\beta_{t,k} \sim \text{Bingham}(\mathbf{C}_{t,k}^{-1})$ .

### B. Model Prediction

Let us say that from time  $t-1$  to  $t$  the radar frame is rotated about its vertical axis by  $\Delta_\psi$  and its position is translated  $[\Delta_x \ \Delta_y]^T$ , then the transformation matrix for the radar coordinate frame is

$$\mathbf{T}_t = \begin{bmatrix} \cos \Delta_\psi & -\sin \Delta_\psi & \Delta_x \\ \sin \Delta_\psi & \cos \Delta_\psi & \Delta_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (5)$$

and a static point is predicted to be transformed in the radar coordinate frame such that

$$\begin{bmatrix} x_{t|t-1} \\ y_{t|t-1} \\ 1 \end{bmatrix} = \mathbf{T}_t^{-1} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ 1 \end{bmatrix}. \quad (6)$$

While the Bingham distribution is only closed under orthonormal transformations, the transition can instead be

applied to the underlying normal distribution. Applying the transformation to given road boundary model coefficients  $\beta_{t-1}$  and applying additive noise  $\mathbf{q}_t \sim \text{Normal}(\mathbf{0}, \mathbf{Q}_t)$  results in

$$\beta_{t|t-1} = \mathbf{F}_t \beta_{t-1} + \mathbf{q}_t \quad (7)$$

where

$$\mathbf{F}_t = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 2(\Delta_x \cos \Delta_\psi + \Delta_y \sin \Delta_\psi) & \cos \Delta_\psi & \sin \Delta_\psi & 0 \\ 2(\Delta_y \cos \Delta_\psi - \Delta_x \sin \Delta_\psi) & -\sin \Delta_\psi & \cos \Delta_\psi & 0 \\ \Delta_x^2 + \Delta_y^2 & \Delta_x & \Delta_y & 1 \end{bmatrix}. \quad (8)$$

Thus the predicted road boundary model becomes

$$\beta_{t|t-1} \sim \text{Bingham}((\mathbf{F}_t \mathbf{C}_{t-1,k} \mathbf{F}_t^T + \mathbf{Q}_t)^{-1}). \quad (9)$$

### C. Variational Inference

While Markov Chain Monte Carlo (MCMC) methods, especially Gibbs sampling, are often used to estimate the posterior of such models, in an online application, the uncertainty of the burn-in period and mixing quality makes their convergence unreliable. Instead, we will use Mean Field Variational Inference (MFVI) in order to estimate the posterior distribution of the DPMM, at a small cost to accuracy in return for much more reliable convergence [22].

For simplicity of notation, let  $\mathbf{B}_t = \{\beta_{t,1}, \dots, \beta_{t,K_t}\}$ ,  $\mathbf{C}_t = \{\mathbf{C}_{t,1}, \dots, \mathbf{C}_{t,K_t}\}$ , and we will momentarily disregard the subscript  $t$  for all variables for the remainder of this subsection. The objective of MFVI then is to find an approximating joint distribution  $q$  of the true joint distribution  $p$  such that

$$q(\mathbf{c}, \boldsymbol{\pi}, \mathbf{B}) \approx p(\mathbf{Z}, \mathbf{c}, \boldsymbol{\pi}, \mathbf{B}), \quad (10)$$

where  $q(\mathbf{c}, \boldsymbol{\pi}, \mathbf{B}) = q(\mathbf{c})q(\boldsymbol{\pi})q(\mathbf{B})$  and  $p(\mathbf{Z}, \mathbf{c}, \boldsymbol{\pi}, \mathbf{B}) = p(\mathbf{Z} | \mathbf{c}, \mathbf{B})p(\mathbf{c} | \boldsymbol{\pi})p(\boldsymbol{\pi})p(\mathbf{B})$ . In practice, this is solved by maximizing the variational lower bound, where for a given factor, the log of the optimal approximating distribution  $q^*$  is known to be proportional to the expected value of the log of the true joint distribution  $p$  over the other factors of the distribution [22]. For example,  $\log q^*(\mathbf{c}) \propto E_{\boldsymbol{\pi}, \mathbf{B}}[\log p(\mathbf{Z}, \mathbf{c}, \boldsymbol{\pi}, \mathbf{B})]$ . For this latent variable model, the result is the Expectation-Maximization (EM) algorithm. In each iteration we will first compute an expectation step (E-Step), followed by a maximization step (M-Step). Iteration proceeds until the parameter estimation converges. We provide a derivation of the E-Step and M-Step in the Appendices of [23].

**E-Step:** We solve for  $q^*(\mathbf{c})$  which gives

$$\gamma_{ik} = E_{\mathbf{c}}[c_i = k] = \frac{\rho_{ik}}{\sum_{j=0}^K \rho_{ij}}, \quad (11)$$

where  $[c_i = k]$  is the Iverson bracket which is equal to one when  $c_i = k$  and zero otherwise, and

$$\rho_{ik} = E_{\pi}[\pi_k] \begin{cases} \mathcal{U}(\mathbf{z}_i; [0, r_{\max}] \times [\theta_{\min}, \theta_{\max}]) & \text{if } k = 0 \\ \mathcal{N}(\hat{\beta}_k^{\top} \phi(\mathbf{z}_i); 0, \sigma_{ik}^2) & \text{if } k > 0 \end{cases} \quad (12)$$

with  $\hat{\beta}_k = E[\beta_k]$  (see end of Sec. IV-A) and  $\mathcal{U}$  and  $\mathcal{N}$  representing uniform and normal probability density functions respectively. Equation (11) is also equivalent to  $p(c_i = k | \mathbf{z}_i)$ .

**M-Step:** We are then able to solve for the other parameters,  $q^*(\boldsymbol{\pi})$  which gives

$$E_{\pi}[\pi_k] = \frac{\alpha_k + \sum_{i=1}^N \gamma_{ik}}{\sum_{j=0}^K (\alpha_j + \sum_{i=1}^N \gamma_{ij})}, \quad (13)$$

and  $q^*(\mathbf{B})$  which gives

$$\beta_k \sim \text{Bingham} \left( \mathbf{C}_k^{-1} + \sum_{i=1}^N \gamma_{ik} \frac{\phi(\mathbf{z}_i) \phi(\mathbf{z}_i)^{\top}}{\sigma_{ik}^2} \right). \quad (14)$$

#### D. Candidate Model Generation

Due to the fact that the MFVI converges towards local optima and not necessarily the global optimum, the quality of the results is heavily dependent upon the initial conditions. Therefore, it is necessary to generate high quality initial estimates of the candidate models to be included in the inference. To accomplish this, we take inspiration from the structure of instantaneous multi-model fitting approaches, especially Progressive-X [24], wherein, a RANSAC model proposal step and a model optimization step are alternately applied. In our case, the model optimization is accomplished via the MFVI and the model proposal is accomplished by a custom RANSAC variant. A single iteration of our RANSAC variant algorithm consists of two steps: sampling and scoring.

**Sampling:** We instantiate a proposal model  $\beta'$  exactly from a sample of 3 detections, i.e. 4 coefficients minus 1 for homogeneity. The random sample is generated from the Reservoir Sampling algorithm [25], wherein each detection  $\mathbf{z}_i$  is included in the sample with probability proportional to  $\gamma_{i0}$  i.e. the probability that  $\mathbf{z}_i$  belongs to the outlier class. Let  $\mathbf{B}' = \mathbf{B}_{t-1} \cup \beta'$ . Additionally, let  $\boldsymbol{\alpha}' = [\boldsymbol{\alpha}_{t-1}^{\top} \ 3]^{\top}$ , then  $\boldsymbol{\pi}' = \frac{\boldsymbol{\alpha}'}{\|\boldsymbol{\alpha}'\|_1}$ , where  $\|\cdot\|_1$  is the L1 norm.

**Scoring:** Given  $\mathbf{B}'$  and  $\boldsymbol{\pi}'$  we compute a single E-Step as described by equation (11) for  $\gamma'_{ik}$ . We define the proposal score as the difference in the expected number of points in the outlier class given the proposal,

$$\xi = \sum_{i=1}^N \gamma'_{i0} - \gamma_{i0}. \quad (15)$$

We maintain the best proposal  $\beta^*$  which has the greatest score  $\xi^*$ .

The confidence of the best proposal at iteration  $j$  is

$$1 - \left( 1 - \left( \frac{\xi^*}{\sum_{i=1}^N \gamma_{i0}} \right)^3 \right)^j, \quad (16)$$

and we terminate the RANSAC algorithm when the confidence is greater than some confidence threshold e.g. 99%. If  $\xi^*$  is greater than some acceptance threshold, then  $\beta^*$  is added to the set of candidate models. The acceptance threshold may be tuned depending upon how conservative we would like to be in adding new candidate models wherein the higher the threshold, the less likely we are to accept the proposal. Generally, this threshold should be greater than the minimum sample size of 3, otherwise it is nearly guaranteed to accept the proposal.

#### E. Algorithm Termination

Given that the view of the scene changes with time, we cannot simply use the posterior model concentration  $\alpha_t$  as the prior for the next radar measurement at timestep  $t + 1$ . Instead, we let the next prior be a moving average of the expected number of points assigned to each model,

$$\alpha_{t,k} = (1 - c)\alpha_{t-1,k} + c \sum_{i=1}^N \gamma_{ik}, \quad (17)$$

where  $c \in [0, 1]$  controls how strongly the current measurement affects the concentration prior wherein  $c = 1$  indicates that the current iteration completely determines the prior for the next iteration and  $c = 0$  indicates that we maintain a constant prior assigned in candidate generation.

Given  $\alpha_{t,k}$ , we eliminate any models which no longer sufficiently contribute to the DPMM. Simply, if  $\alpha_{t,k}$  is below a chosen maintenance threshold, the candidate model  $k$  is removed from the DPMM, otherwise, the values of  $\alpha_{t,k}$ ,  $\mathbf{C}_{t,k}^{-1}$ , and  $\beta_{t,k}$  are used as a prior for the next radar measurement.

Lastly, we choose the candidates to be output as  $\beta_{t,l}$  and  $\beta_{t,r}$ . Given we are attempting to recognize the primary road boundaries to the left and right, we expect them to intercept the radar y-axis. We separate candidates into left and right groups depending on the sign of this y-intercept and select one from each group with y-intercept nearest to the radar origin to be  $\beta_{t,l}$  and  $\beta_{t,r}$  respectively.

#### F. Complexity Analysis

We provide pseudocode of our algorithm with complexity in Alg. 1. We consider that number of candidate models  $K_t$  is bounded by  $K_{\max}$  such that  $K_t \leq K_{\max} \leq \frac{N_t}{D-1}$  in the worst case, where the dimensionality of our model is a constant  $D = 4$ . However, as the road boundaries tend to be consistent in the radar field of view and remain in the candidate set, while other candidate models tend to be more transient in the radar field of view and are removed in time, we find it reasonable in practice to enforce a constant upper bound on the number of candidate models such that  $K_t \leq K_{\max} \ll N_t$  by bypassing the candidate model generation when  $K_t = K_{\max}$ . We note that the number

### Algorithm 1: Road Boundary Estimation

```

Data:  $\alpha_{t-1}, \mathbf{C}_{t-1}$ 
Input:  $\{\mathbf{z}_{t,i}, \Sigma_{t,i}\}_{i=1:N_t}$ 
Output:  $\beta_{t,l}, \beta_{t,r}$ 
// Model Prediction
1  $\mathbf{C}_t |_{t-1}$  from (9)  $\mathcal{O}(K_{t-1})$ 
2  $\mathbf{B}_t |_{t-1} = \{\beta | \min_{\lambda} \mathbf{C}_t^{-1} |_{t-1,k} \beta = \lambda \beta\}$   $\mathcal{O}(K_{t-1})$ 
3 do  $\mathcal{O}(M)$ 
  // Variational Inference
4  $\pi = \alpha_{t-1} / \|\alpha_{t-1}\|_1$   $\mathcal{O}(K_{t-1})$ 
5 do  $\mathcal{O}(V)$ 
  // E-Step
6  $\{\gamma'_{ik}\}$  from (11)  $\mathcal{O}(N_t K_{t-1})$ 
  // M-Step
7  $\pi$  from (13)  $\mathcal{O}(N_t K_{t-1})$ 
8  $\mathbf{C}_t^{-1}$  from (14)  $\mathcal{O}(N_t K_{t-1})$ 
9  $\mathbf{B}_t = \{\beta | \min_{\lambda} \mathbf{C}_t^{-1} \beta = \lambda \beta\}$   $\mathcal{O}(K_{t-1})$ 
10 while not converged
  // Candidate Model Generation
11  $\xi^* = 0, \beta^* = \emptyset, \text{confidence} = 0$   $\mathcal{O}(1)$ 
12 while confidence < confidence threshold do  $\mathcal{O}(G)$ 
  // Sampling
13  $\beta'$  from Reservoir Sample  $\mathcal{O}(\log N_t)$ 
14  $\mathbf{B}' = \mathbf{B}_{t-1} \cup \beta'$   $\mathcal{O}(1)$ 
15  $\alpha' = [\alpha_{t-1}^T \ 3]^T$   $\mathcal{O}(1)$ 
16  $\pi' = \alpha' / \|\alpha'\|_1$   $\mathcal{O}(K_{t-1})$ 
  // Scoring
17  $\{\gamma'_{ik}\}$  from (11) // E-Step  $\mathcal{O}(N_t K_{t-1})$ 
18 if  $\xi$  from (15) >  $\xi^*$  then  $\mathcal{O}(N_t)$ 
19    $\xi^* = \xi, \beta^* = \beta'$   $\mathcal{O}(1)$ 
20   confidence from (16)  $\mathcal{O}(N_t)$ 
21 if  $\xi^* \geq \text{acceptance threshold}$  then  $\mathcal{O}(1)$ 
22    $\mathbf{B}_{t-1} = \mathbf{B}_{t-1} \cup \beta^*, \alpha_{t-1} = [\alpha_{t-1}^T \ 3]^T$   $\mathcal{O}(1)$ 
23 while proposal accepted
  // Algorithm Termination
24  $\alpha_t$  from (17)  $\mathcal{O}(N_t K_{t-1})$ 
25 remove candidate models with  $\alpha_{t,k} < \text{maintenance threshold}$   $\mathcal{O}(K_{t-1})$ 
26  $\beta_{t,l}, \beta_{t,r}$  from  $\mathbf{B}_t$  with minimum left, right lateral distance  $\mathcal{O}(K_t)$ 

```

of candidate models is initially  $K_{t-1}$  and the algorithm proceeds by repeatedly growing the initial candidate set before reducing it to its final size of  $K_t$  during algorithm termination, so to aid our analysis, we will consider the complexity in terms of  $K_{\max}$  rather than at any particular time  $t$ .

The model prediction stage takes a constant number of operations for each candidate model and thus a total of  $\mathcal{O}(K_{\max})$  operations. Let the maximum number of iterations for convergence of the variational inference be  $V$ . Both the E-Step and the M-Step take  $\mathcal{O}(N_t K_{\max})$  operations, thus each round of variational inference takes  $\mathcal{O}(V N_t K_{\max})$  operations. Let the maximum number of iterations of candidate generation be  $G$ . Generating a minimum sample from reservoir sampling take  $\mathcal{O}(\log N_t)$  operations for a fixed reservoir size [25], and scoring, primarily consisting of a single E-Step, takes  $\mathcal{O}(N_t K_{\max})$  operations. Thus, each round of candidate generation takes  $\mathcal{O}(G N_t K_t)$  operations. Let the maximum number of iterations of the main loop of our algorithm, consisting of variational inference and candidate generation be  $M \leq K_{\max} - K_{t-1}$ , then the total number of operations for this main loop is  $\mathcal{O}(M(V + G)N_t K_{\max})$ . Finally, algorithm termination takes  $\mathcal{O}(N_t K_{\max})$  operations. Therefore, the total runtime at time  $t$  of our algorithm is on the order of  $\mathcal{O}(M(V + G)N_t K_{\max})$  which reduces to  $\mathcal{O}(M(V + G)N_t^2)$  in the worst case or  $\mathcal{O}((V + G)N_t)$  if

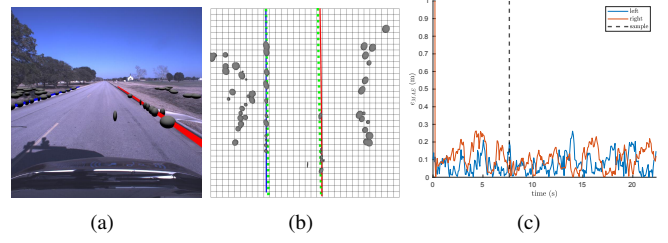


Fig. 3. Results from Trajectory 1. (a) Sample camera view and (b) sample top-down orthographic view of scene where grey ellipses are the radar target detections, green squares are ground truth boundary points, and the blue and red curves are the estimated left and right boundaries respectively. Grid has 1 meter spacing. (c) The MAE of the estimated road boundary models vs time, where the dashed line represent the time of the sample in (a) and (b).

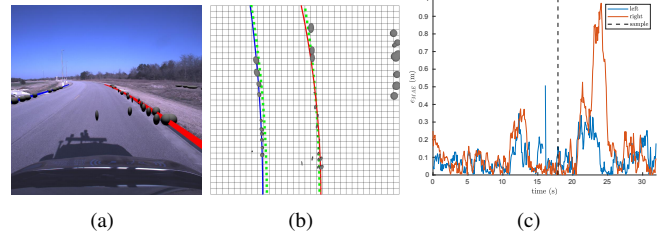


Fig. 4. Results from Trajectory 2. (a) Sample camera view. (b) Sample top-down orthographic view. (c) MAE vs time over the trajectory.

we apply a more reasonable constant upper bound on the number of candidate models.

One can tune this computational complexity of the algorithm in a few ways. An ideal selection of  $K_{\max}$  will keep the candidate set saturated and ideal selection of the proposal acceptance and candidate maintenance thresholds will restrict the algorithm to only consider candidates with sufficient support which together reduce the number of main loop iterations  $M$ .  $V$  may be reduced by lowering the convergence threshold of the variational inference, though we note that given a smooth transition between timesteps, the variational inference is often near convergence a priori via the application of the predictive model.  $G$  may be similarly reduced by setting a lower confidence threshold on candidate model generation. Finally, because the main loop iterations monotonically increase the variational lower bound, one can proceed to algorithm termination at any point (e.g. by time or iteration threshold) to retrieve a so-far-the-best estimate.

The memory consumption of our algorithm is dominated by the two  $N_t \times K_{\max}$  candidate-detection association matrices with elements  $\gamma_{ik}$  and  $\gamma'_{ik}$  generated during the E-Steps in variational inference and candidate generation respectively. This results in a memory consumption on the order of  $\mathcal{O}(N_t^2)$  in the worst case or  $\mathcal{O}(N_t)$  if we apply the constant upper bound on the number of candidate models.

## V. EXPERIMENTS

The proposed algorithm is developed as a Robotic Operating System (ROS) module in C++. We test the algorithm on data collected from a Continental ARS430 automotive Doppler radar, and the algorithm is validated against ground truth measurements of the boundary location represented as

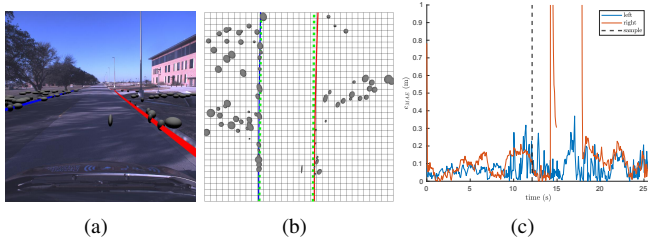


Fig. 5. Results from Trajectory 3. (a) Sample camera view. (b) Sample top-down orthographic view. (c) MAE vs time over the trajectory.

discrete points along the boundary with approximately 1 meter spacing. The vehicle drives each trajectory at roughly the posted speed limit of the given roadway.

In order to quantify the results of our algorithm, at each time step we compute the Mean Absolute Error (MAE) of the estimated left and right road boundaries relative to their associated ground truth points. Let us consider a given estimated road boundary  $\beta$  and an associated set of ground truth points  $\mathbf{S} = \{s_1, \dots, s_N\}$ . We find that the ground truth points are somewhat conservative, being approximately 20 cm closer to the interior of the roadway than the radar measured points. Therefore, in order to account for this and any GPS offset, we first calculate the mean error over the entire trajectory, which we consider to be the bias due to the ground truth and GPS offsets. The mean error of the estimated boundary is  $e = \frac{1}{N} \sum_{i=1}^N d(s_i, \beta)$  where  $d(s_i, \beta)$  is the signed geometric distance of the ground truth point  $s_i$  from the boundary  $\beta$ . We then compute the mean error  $\bar{e}$  and error standard deviation  $\sigma_e$  over the entire trajectory, excluding any individual estimates where  $|e - \bar{e}| > 3\sigma_e$ , which we consider to be failure cases. The mean absolute error of each individual estimate is then  $e_{\text{MAE}} = \frac{1}{N} \sum_{i=1}^N |d(s_i, \beta) - \bar{e}|$ . Additionally, we calculate the failure rate for the estimation of each the left and right boundaries to be the ratio of the number of timesteps where  $|e - \bar{e}| > 3\sigma_e$  or no estimate is given to the total number of timesteps.

We demonstrate three different trajectories where we have ground truth data, for which summarized results are presented in Table I. Trajectory 1, demonstrated in Fig. 3, is along a straight, curbed roadway with minimal clutter near the boundary. Similarly, Trajectory 2, demonstrated in Fig. 4, is along a curbed roadway with minimal clutter near the boundary, but transitions between straight and curved segments. Trajectory 3, demonstrated in Fig. 5, presents a more challenging case along a straight, curbed roadway, featuring clutter on the left roadside, obfuscating the boundary, as well as a driveway intersecting the right boundary.

We consider Trajectory 1 to be the most ideal case for our algorithm, and, consequently, it produces the most stable estimation with no failure cases outside the first few timesteps as the estimation stabilizes. Trajectory 2 results in the lowest estimation accuracy, largely due to the road curvature transitions at  $t \approx 11$  s and  $t \approx 21$  s which temporarily violate Assumption 3. However, even in this case, the estimation is not considered to fail. For Trajectory 3, the mean MAE

TABLE I  
SUMMARIZED RESULTS

Trajectory	Boundary	$\bar{e}_{\text{MAE}}$ (cm)	$\sigma_{e_{\text{MAE}}}$ (cm)	% Failure
1 (300 m)	left	7.44	8.70	0
	right	10.7	6.68	0.96
2 (500 m)	left	9.36	11.9	1.13
	right	11.0	10.2	7.98
3 (220 m)	left	7.50	9.17	0.56
	right	9.98	11.7	14.0

is roughly consistent with the ideal case, but with increased variance. Most generally, the estimation accuracy is typically better for the left boundary than the right boundary. This is due to the smaller angle of incidence to the left boundary, providing a greater number of detections. We demonstrate that the algorithm is capable of successful estimation under various challenging conditions in Figs. 1 and 6.

Finally, we identify the common failure modes of the algorithm. The most common, demonstrated in Fig. 7(a), occurs when non-boundary detections with high leverage cause the estimation to deviate at long range. This occurs in Trajectory 2 at  $t \approx 22$  s. The other common failure mode, demonstrated in Fig. 7(b), occurs when the primary boundary deviates to join with the boundary of an intersecting road or driveway. This occurs in Trajectory 3 at  $t \approx 14$  s. Lastly, the boundary can be occluded by passing vehicles, which occurs once in Trajectory 2 at  $t \approx 15$  s when a vehicle passes to the left. It is worthwhile to note that in these failure modes, while the estimation is incorrect at long range, the estimation is often still reasonable at short range.

## VI. CONCLUSION

To provide a fallback solution for autonomous vehicles under severe weather conditions, we presented a novel road boundary detection method that is solely based on a sparse radar point cloud. The method was built on a homogeneous boundary model and we derived a probability distribution of road boundary models based on the radar point cloud using variational inference. In order to generate initial candidate models, we developed a custom RANSAC variant to propose unseen model instances as candidate road boundaries. By alternating variational inference and RANSAC proposals until convergence we generated the best estimate of all candidate models. We selected the candidate model with the minimum lateral distance to the radar on each side as the estimates of the left and right road boundaries. The algorithm has been implemented as a ROS module and tested under real radar data. The results are satisfactory.

In the future, we will investigate global representations for road boundary mapping and develop a sensor-fusion approach to further improve robustness.

## ACKNOWLEDGEMENT

The authors thank Di Wang, Shuangyu Xie, and Fengzhi Guo for their input and feedback.

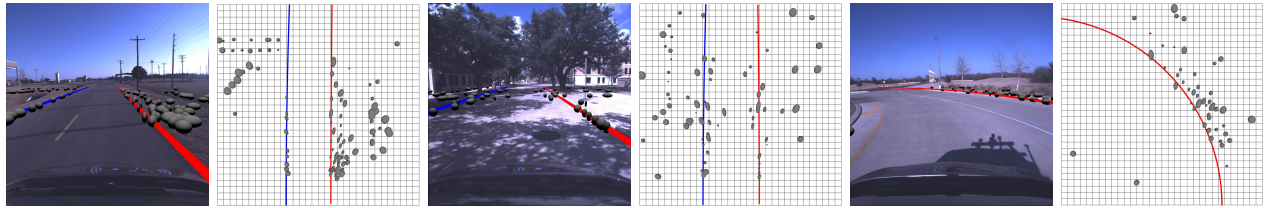


Fig. 6. A sample of successful road boundary estimations under various challenging scenarios. left: natural road boundaries without curbs, middle: significant clutter in an urban scene, right: through a roundabout, note that the left boundary is not detectable due to its curving out of the radar field of view.

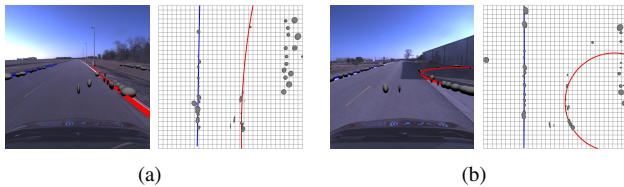


Fig. 7. A sample of common failure modes. (a) High leverage points from clutter can disrupt the boundary estimation at long range. (b) Deviation of the boundary at an intersection or driveway.

## REFERENCES

- [1] A. Srivastav and S. Mandal, "Radars for autonomous driving: A review of deep learning methods and challenges," *IEEE Access*, vol. 11, pp. 97147–97168, 2023.
- [2] M. Serfling, R. Schweiger, and W. Ritter, "Road course estimation in a night vision application using a digital map, a camera sensor and a prototypical imaging radar system," in *2008 IEEE Intelligent Vehicles Symposium*, pp. 810–815, IEEE, 2008.
- [3] D. Patel and K. Elgazzar, "Road boundary detection using camera and mmwave radar," in *2022 5th International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, pp. 1–7, IEEE, 2022.
- [4] F. Homm, N. Kaempchen, J. Ota, and D. Burschka, "Efficient occupancy grid computation on the gpu with lidar and radar for road boundary detection," in *2010 IEEE intelligent vehicles symposium*, pp. 1006–1013, IEEE, 2010.
- [5] L. M. Romero, J. A. Guerrero, and G. Romero, "Road curb detection: a historical survey," *Sensors*, vol. 21, no. 21, p. 6952, 2021.
- [6] M. Nikolova and A. Hero, "Segmentation of a road from a vehicle-mounted radar and accuracy of the estimation," in *Proceedings of the IEEE Intelligent Vehicles Symposium 2000 (Cat. No.00TH8511)*, pp. 284–289, 2000.
- [7] K. Kaliyaperumal, S. Lakshmanan, and K. Kluge, "An algorithm for detecting roads and obstacles in radar images," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 1, pp. 170–182, 2001.
- [8] K.-Y. Guo, E. G. Hoare, D. Jasteh, X.-Q. Sheng, and M. Gashinova, "Road edge recognition using the stripe hough transform from millimeter-wave radar images," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 2, pp. 825–833, 2014.
- [9] K. Werber, J. Klappstein, J. Dickmann, and C. Waldschmidt, "Association of straight radar landmarks for vehicle self-localization," in *2019 IEEE Intelligent Vehicles Symposium (IV)*, pp. 736–743, IEEE, 2019.
- [10] C. Lundquist, T. B. Schön, and U. Orguner, "Estimation of the free space in front of a moving vehicle," 2009.
- [11] F. Xu, H. Wang, B. Hu, and M. Ren, "Road boundaries detection based on modified occupancy grid map using millimeter-wave radar," *Mobile Networks and Applications*, vol. 25, no. 4, pp. 1496–1503, 2020.
- [12] T.-Y. Lee, V. Skvortsov, M.-S. Kim, S.-H. Han, and M.-H. Ka, "Application of  $w$ -band fmcw radar for road curvature estimation in poor visibility conditions," *IEEE Sensors Journal*, vol. 18, no. 13, pp. 5300–5312, 2018.
- [13] C. Lundquist, L. Hammarstrand, and F. Gustafsson, "Road intensity based mapping using radar measurements with a probability hypothesis density filter," *IEEE Transactions on Signal Processing*, vol. 59, no. 4, pp. 1397–1408, 2011.
- [14] A. Kingery and D. Song, "Improving ego-velocity estimation of low-cost doppler radars for vehicles," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9445–9452, 2022.
- [15] Virginia Sheriffs' Association, "Thunderclap highway safety message – march 23 at 3 p.m.," <https://vasheriff.org/2021/03/17/thunderclap-highway-safety-message-march-23-at-3-p-m/>. (accessed Feb. 28, 2023).
- [16] F. Faion, A. Zea, and U. D. Hanebeck, "Reducing bias in bayesian shape estimation," in *17th International Conference on Information Fusion (FUSION)*, pp. 1–8, IEEE, 2014.
- [17] Z. Zhang, "Parameter estimation techniques: A tutorial with application to conic fitting," *Image and vision Computing*, vol. 15, no. 1, pp. 59–76, 1997.
- [18] C. Bingham, "An antipodally symmetric distribution on the sphere," *The Annals of Statistics*, pp. 1201–1225, 1974.
- [19] I. Gilitschenski, G. Kurz, S. J. Julier, and U. D. Hanebeck, "Unscented orientation estimation based on the bingham distribution," *IEEE Transactions on Automatic Control*, vol. 61, no. 1, pp. 172–177, 2015.
- [20] K. Li, D. Frisch, B. Noack, and U. D. Hanebeck, "Geometry-driven deterministic sampling for nonlinear bingham filtering," in *2019 18th European Control Conference (ECC)*, pp. 381–387, IEEE, 2019.
- [21] W. Wang and P. G. Adamczyk, "Comparison of bingham filter and extended kalman filter in imu attitude estimation," *IEEE Sensors Journal*, vol. 19, no. 19, pp. 8845–8854, 2019.
- [22] D. M. Blei and M. I. Jordan, "Variational inference for Dirichlet process mixtures," *Bayesian Analysis*, vol. 1, no. 1, pp. 121 – 143, 2006.
- [23] A. Kingery and D. Song, "Road boundary estimation using sparse automotive radar inputs," *arXiv*, 2023.
- [24] D. Barath and J. Matas, "Progressive-x: Efficient, anytime, multi-model fitting algorithm," in *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3780–3788, 2019.
- [25] P. S. Efraimidis and P. G. Spirakis, "Weighted random sampling with a reservoir," *Information processing letters*, vol. 97, no. 5, pp. 181–185, 2006.