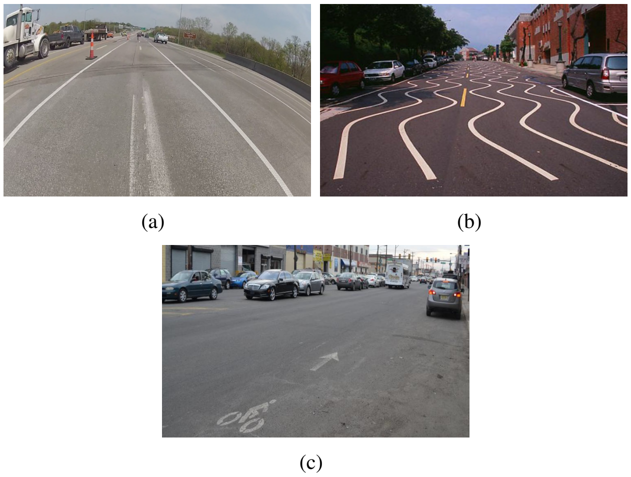
Lane Marking Quality Assessment for Autonomous Driving

Binbin Li, Dezhen Song, Haifeng Li, Adam Pike, and Paul Carlson

***Abstract*— Measuring the quality of roads and ensuring they are ready for autonomous driving is important for future transportation systems. Here we focus on developing metrics and algorithms to assess lane marking qualities from an egocentric view of an inspection vehicle equipped with a global positioning system (GPS) receiver, a frontal-view camera, and a light detection and ranging (LIDAR). We propose three quality metrics for lane markings: correctness, shape, and visibility. The correctness metric measures the divergence between the expected lane markings based on prior map inputs and the actual sensor inputs. The shape metric evaluates smoothness in road curvature and width range. The visibility metric evaluates the contrast between lane markings and background road surfaces. We establish the metrics using probability modeling by considering uncertainties in prior maps and sensory inputs. Fusing camera images and LIDAR data, we propose a dual-modal algorithm to compute these metrics. We have implemented the algorithms and tested them under KITTI dataset. The results show that our metrics can successfully detect lane marking anomalies in all testing scenarios.**



(a)

(b)

(c)

Fig. 1. Examples of urban road scenario. a) New lane markings coexists with the old one. b) Lane border shape may not satisfy the standard. c) Faded or blurred lane markings.

I. Introduction

As autonomous vehicles (AV) are getting closer and closer to our life, one critical question remains unanswered: are our roads ready for AVs? AV developers attempt to deal with all kinds of road conditions. However, safety can be challenged when poor road conditions (see Fig. 1) appear because there are limited training data for abnormalities. Due to the limited sensory capabilities and on-board computation resources of AVs, exhaustively predicting road scenarios is infeasible. A solution to reliable autonomous driving is to ensure our infrastructure is ready for the technology.

Here we present a lane marking quality assessment (LMQA) method to help road inspection crews examine the quality of lane markings. The method assumes a vehicle egocentric view with a global positioning system (GPS) receiver, a frontal-view camera, and a light detection and ranging (LIDAR). Based on data from GPS, prior maps from geographic information systems (GIS) and on-board sensors, we propose three different lane quality metrics: *correctness*, *shape*, and *visibility*. The correctness metric measures the divergence between the expected lane markings based on prior map inputs and the actual sensor inputs. Building on the difference between posterior distributions, it takes uncertainties in inputs into consideration. The shape metric verifies if the lane has smooth curvature according to road grade and is within satisfying width range. The visibility metric evaluates the contrast between lane markings and background road surfaces. Fusing camera images and LIDAR data, we propose an algorithm to compute these metrics. The algorithms extensively utilize both camera images and LIDAR data in road surface detection and lane marking recognition to generate segmented left and right lane marking points required by the three metrics computation. We have implemented the algorithms and tested it using KITTI dataset. The results show that our metrics successfully recognize anomalies in lane markings.

It is worth noting that LMQA is NOT the same as the well known lane marking detection problem. It is not our best interest to develop/apply the most sensitive and accurate lane detection algorithm in LMQA because we want to ensure that our roads are safe for less capable vehicles. *Here we measure roads instead of vehicles*. LMQA focuses on evaluating lane marking qualities instead of abilities to detect them. It needs to be able to mark low quality road segments and output different types of quality issues instead of just reporting "no lane" detected. It also takes into consideration of common sensor configurations for all AVs instead of optimizing lane detection for a particular sensor configuration.

B. Li and D. Song are with Department of Computer Science and Engineering, Texas A&M University, College Station, TX, 77843, US. (email: binbinli@tamu.edu, dzsong@cse.tamu.edu)

H. Li is with Computer Science Department, Civil Aviation University of China, Tianjin, 300300, China, and also with Fujian Provincial Key Laboratory of Information Processing and Intelligent Control, Minjiang University, Fuzhou, 350108, China. (email:hfli@cauc.edu.cn)

A. Pike is with Texas A&M Transportation Institute, College Station, TX, 77843, US. (email: A-Pike@tti.tamu.edu)

P. Carlson is with Road Infrastructure, Inc., College Station, TX, 77843, US. (email: PCarlson@roadinfrastructureinc.com)

This work was supported in part by National Science Foundation under IIS-1318638, NRI-1426752, and NRI-1526200, and in part by TxDot 0-6869.

II. Related Work

Our LMQA research is related to road quality assessment, road surface extraction, and lane detection in transportation and autonomous driving research.

Lane markings play an important role in autonomous driving. To the best of our knowledge, little has been done to quantify lane markings to assist road maintenance. Harwood et al. [1] use operational analysis procedures to assess the capacity and level of service of two-lane highways. Flannery et al. [2] quantify the road service quality by comparing drivers' assessments of the performance of urban streets with objective measures of performance. Thomas et al. [3] propose a systematic approach to evaluate algorithms for extracting road marking features. Pohl et al. [4] estimate driver's visual distraction level to provide sufficient reliability of lane-keeping. However, none of them focus on measuring lane marking quality itself.

During road surface extraction, color and texture are the main perceptual cues for navigation systems of semi- or fully autonomous vehicles. Besides obstacle avoidance, road surface detection facilitates path planning and decision making. Common sensory methods include cameras [5] and LIDARs. Pradeep et al. [6] use stereo camera data to extract the road surface structure. Hernández et al. [7] filter and segment the road surface from 3D point clouds acquired through mobile LIDAR systems. Li et al. [8] utilize the structural information to find the road surface by combining multiple task deep convolutional neural networks with a recurrent neural network detector. Yu et al. [9] extract road surface points directly from three dimensional point clouds. Guan et al. [10] extract road surface through a curb-based method using geo-referenced intensity images. More detailed surveys can be found in [11], [12]. Most existing efforts only utilize a single sensing modality, as we simultaneously employ both camera images and LIDAR data to extract the road surface in our approach for more robustness.

A lane border detection system detects lane markings from complex environments. Lane markings are important for reliable estimation of vehicle positions relative to lanes. Different sensors or perception modalities have been used for lane border detection, such as monocular vision [11], [13], LIDAR [14]–[16], stereo imaging [6], [17], GPS and inertial measurement unit (IMU) [18], [19], and Radar [20]. Gu et al. [21] classify the lane markings by fusing images and LIDAR scans using convolutional neural networks. Huang et al. [22] describe and detect multiple lane borders in an urban road network from calibrated video imagery and laser range data acquired by a moving vehicle. Mammeri et al. [23] combine the Maximally Stable Extremal Region technique with the Hough Transform to detect and recognize lane markings. Various lane border detection systems have been proposed in the automotive industry [24], [25]. Built on existing efforts, our dual-modal lane marking detection leverages inputs from both camera images and LIDAR scans to facilitate lane marking quality metric computation with an attempt to provide a baseline performance under common sensory configuration.

III. Problem Formulation

A. Inputs, Assumptions, and Notations

Our objective is to quantify the lane marking quality from a vehicle egocentric view. The inspection vehicle is equipped with a frontal view camera and a LIDAR, and we use their data as problem input. We also employ GPS data and prior maps consisting of a set of 3D lane boundaries, such as Google Maps, as part of inputs. We only evaluate immediate left and right lane markings with respect to the vehicle due to their importance in guiding the vehicle. We do not evaluate multiple parallel lanes simultaneously because the sensors on-board the vehicle have perspective limitations. We have the following assumptions.

a.1 The camera is pre-calibrated and we know its intrinsic parameters. The nonlinear distortion of camera images has been removed.

a.2 The GPS, the camera, and the LIDAR readings are already synchronized.

a.3 The coordinate system transformations between any two sensors are known by calibration.

Common notations are defined as follows,

- $\mathbf{P}_{i,t} \in \mathbb{R}^3$, the i-th 3D LIDAR point in the LIDAR reference system at time t. Also, it is defined in a system with x-axis pointing to vehicle forward direction, y-axis pointing to the left of the vehicle lateral direction, and z-axis pointing upward.
- I_i, the intensity value of the LIDAR point $\mathbf{P}_{i,t}$.
- $\mathcal{P}_t := \{\mathbf{P}_{i,t}\}$, LIDAR point cloud data set at time t.
- $\mathbf{I}_t$, the gray-scale camera image at time t.
- $\mathbf{p}_{k,t} = [u\ v]^\intercal \in \mathbb{R}^2$, the k-th pixel point in image $\mathbf{I}_t$ where (u, v) is the image coordinate.
- $\tilde{\mathbf{X}}$, homogeneous vector $\tilde{\mathbf{X}} = [\mathbf{X}^\intercal, 1]^\intercal$ where $\mathbf{X}^\intercal$ denotes the inhomogeneous part of $\tilde{\mathbf{X}}$.

B. Quality Metrics and Problem Definition

We introduce three types of quality metrics for LMQA. Here we just define them. We will model them mathematically in Section IV.

- *Correctness Metric:* Defined as μ_c, this metric quantifies the divergence between sensed lane marking positions and that from the prior map in GIS system. It may be caused by slow updates of GIS database when road construction or maintenance changes lane markings. Lane markings may disappear completely or painted incorrectly due to poor maintenance. All of these cause discrepancies between prior maps and sensory inputs which introduces difficulty for AVs to make decisions.
- *Shape Metric:* Defined as μ_s, this metric measures whether the road segment is smooth in curvature which is defined by road grade and conformity to lane width standard. A smooth road with proper width makes it easier for AVs to perform trajectory following and leads to smooth and safe rides. Lane width may differ according to different road grades and countries but is

always bounded between a lower bound and an upper bound. Sometimes, the desired width may be a single fixed value. For example, the US Interstate Highway standard dictates 3.7-meter lane width.

- *Visibility Metric:* Defined as μ_v, the metric measures how visible lane markings compare to background surface in both images and LIDAR data. High contrast makes lane markings easy to be detected and segmented by AVs.

With assumptions, notations and quality metrics made, our problem can be defined as follows,

Problem 1: Given the GPS coordinate, a prior map, the LIDAR point cloud $\mathcal{P}_t$, and camera image $\mathbf{I}_t$, quantify the LMQA according to the aforementioned quality metrics.

IV. Metric Modeling

Now let us model the three metrics mathematically. Note that we sample data periodically. At discrete time t, we have a camera image $\mathbf{I}_t$ and LIDAR point cloud $\mathcal{P}_t$. To avoid too much overlap with previous or following iterations, we only use a partial set of points

$$\bar{\mathcal{P}}_t := \{\mathbf{P}_{i,t} | \|\mathbf{P}_{i,t}\| \leq \zeta \cdot v\tau, \mathbf{P}_{i,t} \in \mathcal{P}_t\}, \tag{1}$$

where $\|\cdot\|$ is the vector l^2-norm, τ is the sampling interval, v is current vehicle velocity, and ζ is a positive constant controlling the overlap between $\bar{\mathcal{P}}_t$ and $\bar{\mathcal{P}}_{t+1}$. To ensure full coverage, we set $\zeta = 2$. Since we know the relationship between the image coordinate and LIDAR coordinate, we also use $\bar{\mathcal{P}}_t$ to obtain the corresponding $\bar{\mathbf{I}}_t$ in $\mathbf{I}_t$. Also, given the GPS coordinate, we know the prior map region overlaps with $\bar{\mathcal{P}}_t$. Define $\mathbf{X}_p \in \mathbb{R}^3$ as the corresponding lane marking from the prior map in this overlapping region and set $\mathcal{P}_p := \{\mathbf{X}_p\}$ for all $\mathbf{X}_p$'s. All metrics below are based on $\mathcal{P}_p$, $\bar{\mathcal{P}}_t$, and $\bar{\mathbf{I}}_t$.

A. Correctness Metric

We define correctness metric in $\bar{\mathbf{I}}_t$. Let C_k represent an event that a pixel $\mathbf{p}_{k,t}$ in image $\bar{\mathbf{I}}_t$ is a lane marking pixel,

$$C_k = \begin{cases} 1, & \mathbf{p}_{k,t} \text{ is a lane marking point} \\ 0, & \text{otherwise.} \end{cases}$$

Define $P(C_k)$ as the probability for event C_k. Define prior map lane pixel $\mathbf{x}_p$ as the projection of $\mathbf{X}_p$ from the prior map into $\bar{\mathbf{I}}_t$. Define $\mathbf{K}$ as the intrinsic camera parameters and $\{\mathbf{R}, \mathbf{t}\}$ as the extrinsic parameters between the camera and LIDAR, where $\mathbf{R}$ and $\mathbf{t}$ are the rotation matrix and translation vector that relate the laser coordinate system to the camera coordinate system. The projection is based on perspective projection model,

$$\tilde{\mathbf{x}}_p = \mathbf{K}[\mathbf{R} \;\; \mathbf{t}]\tilde{\mathbf{X}}_p. \tag{2}$$

Define $\mathcal{S}_P = \{\mathbf{x}_p\}$ to be the set that covers all points in $\mathcal{P}_p$. Define set $\mathcal{S}_Q$ that contains all the lane marking pixels in $\bar{\mathbf{I}}_t$. Define posterior probability $P(C_k|\mathcal{S}_P)$ to capture the lane marking distribution in the image space through a prior map. It is not deterministic because we have uncertainties in the map due to resolution limitations and errors in GPS coordinates. Similarly, we define posterior probability $P(C_k|\mathcal{S}_Q)$ to be the lane marking distribution given the sensory inputs. It is probabilistic due to sensory uncertainty. Then the correctness metric is modeled as the difference between these two conditional probabilistic distributions. We employ the smoothed Kullback-Leibler (KL) divergence to characterize this difference,

$$\mu_c = \sum_k P(C_k|\mathcal{S}_P) \log \frac{P(C_k|\mathcal{S}_P)}{P(C_k|\mathcal{S}_Q)}, \tag{3}$$

A KL divergence of 0 indicates that we have two identical distributions, while a KL divergence of 1 indicates that the two distributions are totally different. Therefore, we prefer small values in this metric.

Now let us explain how to compute $P(C_k|\mathcal{S}_P)$ and $P(C_k|\mathcal{S}_Q)$. Recall that $\mathbf{X}_p \in \mathcal{P}_p$ represents a lane marking point in the prior map and $\mathbf{x}_p \in \mathcal{S}_P$ is its projection using (2). We use the set $\{(\mathbf{x}_p, C_p)\}$ as the training set to instantiate a recursive Bayesian estimation process to obtain the lane marking distribution $P(C_k|\mathcal{S}_P)$ for the lane information on the prior map. It can be computed by using a two-phase approach. For an $\mathbf{x}_p^* \in \mathcal{S}_P$ and its corresponding event C_p^*, we update the probability distribution,

$$P(C_q^*|\mathcal{S}_P) = \overline{bel}(C_q^*|f(\mathbf{x}_q^*), \sigma^2, \mathbf{x}_q^*, \mathcal{S}_P), \tag{4}$$

where σ^2 is the variance of the noise, and the latent function f is represented by Gaussian Process (GP) [26]. The GP provides posterior distribution of pixel $\mathbf{x}_k \in \bar{\mathbf{I}}_t$ for prediction,

$$P(C_k|\mathcal{S}_P) = \overline{bel}(C_k|\mu_I, \sigma_I^2, \mathbf{x}_k, \mathcal{S}_P). \tag{5}$$

Here, μ_I and σ_I^2 are the expectation and variance of the posterior distribution related to the kernel function, which characterizes the correlation between the function values at different pixels. Here we employ a Gaussian kernel K as

$$K(\mathbf{p}_{i,t}, \mathbf{p}_{j,t}) = \sigma_f^2 \exp(-\frac{1}{2\lambda_f^2}\|\mathbf{p}_{i,t} - \mathbf{p}_{j,t}\|^2), \tag{6}$$

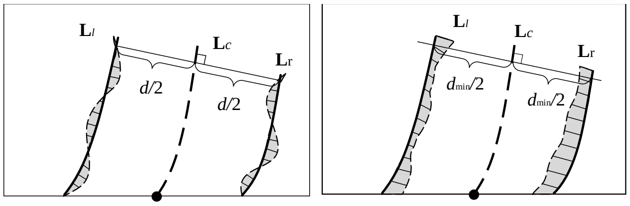
for $\{\mathbf{p}_{i,t}, \mathbf{p}_{j,t}\} \subset \bar{\mathbf{I}}_t$ with $\mu_I = \mathbf{k}_*^\intercal(\mathbf{K}_o + \sigma^2\mathbf{I})^{-1}C_p^*$ and $\sigma_I^2 = \mathbf{k}_{**}\mathbf{k}_*^\intercal(\mathbf{K}_o + \sigma^2\mathbf{I})^{-1}\mathbf{k}_*$ where σ_f is the variance of the lane marking position, λ_f is the length scale variable, $\mathbf{k}_* = K(\mathcal{S}_P, \mathbf{x}_k)$, $\mathbf{K}_o$ is the kernel matrix of the training data $\mathcal{S}_P$, $\mathbf{k}_{**} = K(\mathbf{x_k}, \mathbf{x_k})$, and $\mathbf{I}$ is an identity matrix.

Similarly, we get $P(C_k|\mathcal{S}_Q)$ through (4)-(5). Therefore, we can obtain μ_c in (3) through the two posterior distributions $P(C_k|\mathcal{S}_Q)$ and $P(C_k|\mathcal{S}_P)$.

B. Shape Metric

The shape metric evaluates the lane shape defined by lane boundaries by examining its smoothness and width. To achieve this, we need to find the 'best shape' that fits the observation data. The best shape refers to a segment of lane that has smooth curvature defined by road grade and fits the width requirement. Then we evaluate how the existing lane marking point set compares to the best shape.

First, we need to model the best shape. We adopt a cubic polynomial lane center curve with a width to describe it because a cubic polynomial is sufficient to describe road



(a) (b)

Fig. 2. An illustration of how to compute lane center curve and the shape metric. Solid curves correspond to $\mathbf{L}_l$ and $\mathbf{L}_r$ which are determined by $\mathbf{L}_c$. Thin and dashed curves close to them represent points $\mathbf{X}_l$ and $\mathbf{X}_r$, respectively. The area of shaded region is μ_s. (a) A cubic polynomial fitting smooths observation noises. (b) An overly narrow lane results in large μ_s.

curve and can be computed straightforwardly from cubic spline fitting. Define the lane center curve as a function of univariate parameter s,

$$\mathbf{L}_c(s) = \mathbf{a}_0 + \mathbf{a}_1 s + \mathbf{a}_2 s^2 + \mathbf{a}_3 s^3 \in \mathbb{R}^3, \tag{7}$$

where $\{\mathbf{a}_i | i = 0, 1, 2, 3\}$ are 3-vectors for polynomial coefficients, and $\mathbf{p}_a = [\mathbf{a}_0^\mathsf{T}, \mathbf{a}_1^\mathsf{T}, \mathbf{a}_2^\mathsf{T}, \mathbf{a}_3^\mathsf{T}]^\mathsf{T}$. By forcing $\mathbf{L}_c(s) \subset \mathcal{P}_t \cup \mathcal{P}_{t-1}$, we obtain s range set $\mathbf{S}$:

$$\mathbf{S} := \{s | \mathbf{L}_c(s) \subset \mathcal{P}_t \cup \mathcal{P}_{t-1}\}.$$

The reason that we choose s's range, $\mathbf{S}$, to be much bigger than that of $\mathcal{P}_t$ is to ensure smoothness in future curve transition and full lane boundary coverage. For each point $\mathbf{X}_c \in \mathbf{L}_c(s)$ and a given width d, we find a point on the left boundary and a point on the right boundary by walking along the direction perpendicular to $\mathbf{L}_c(s)$ by $d/2$ to the left or right, respectively (see Fig. 2(a)). Therefore, lane boundaries $\mathbf{L}_l(s)$ and $\mathbf{L}_r(s)$ are determined by $\mathbf{L}_c(s)$ for the given width d.

Now let us explain how to obtain the lane center curve. Define $\mathbf{X}_l$ and $\mathbf{X}_r$ to be left and right lane marking points in $\bar{\mathcal{P}}_t$, respectively. For a given left boundary set $\mathbf{L}_l(s)$, we evaluate each point in $\mathbf{X}_l$ by measuring the closest distance d_l:

$$d_l(\mathbf{X}_l, \mathbf{L}_l) = \min_{s \in \mathbf{S}} \|\mathbf{X}_l - \mathbf{L}_l(s)\| \approxeq \min_{s \in \mathbf{S}} \|\mathbf{X}_l - \mathbf{L}_c(s)\| - d/2. \tag{8}$$

The approximation works when the road is relatively flat. Similarly, we can evaluate each point in $\mathbf{X}_r$ by measuring the closest distance d_r:

$$d_r(\mathbf{X}_r, \mathbf{L}_l) = \min_{s \in \mathbf{S}} \|\mathbf{X}_r - \mathbf{L}_r(s)\| \approxeq \min_{s \in \mathbf{S}} \|\mathbf{X}_r - \mathbf{L}_c(s)\| - d/2.$$

It is clear that both left and right boundaries are based on the lane center curve which needs to be estimated with respect to inputs $\mathbf{X}_l$ and $\mathbf{X}_r$. We evaluate a given lane center curve using the observations $\mathbf{X}_l$ and $\mathbf{X}_r$ by the following objective function,

$$f_l(\mathbf{L}_c, d) = \frac{1}{n_l} \sum_{\mathbf{X}_l \in \bar{\mathcal{P}}_t} d_l(\mathbf{X}_l, \mathbf{L}_l) + \frac{1}{n_r} \sum_{\mathbf{X}_r \in \bar{\mathcal{P}}_t} d_r(\mathbf{X}_r, \mathbf{L}_r), \tag{9}$$

where n_l and n_r are numbers of lane marking points in the left and right lanes, respectively. Therefore, we can obtain the lane center curve and the optimal width by minimizing the

$$[\mathbf{p}_a^{*T},\ d^*]^{\mathrm{T}} = \arg\min_{\mathbf{p}_a, d} f(\mathbf{L}_c, d), \tag{10}$$

subject to width constraint

$$d_{\min} \leq d \leq d_{\max}, \tag{11}$$

where $\mathbf{p}_a^*$ and d^* are optimal lane center curve parameters and the optimal lane width, and $d_{\min}$ and $d_{\max}$ are the minimum and maximum allowable width, respectively. The $\mathbf{p}_a^*$ defines the optimal center curve L_c^* according to (7), and can be further constrained to reflect desirable curvature range according to the road grade from GIS information so that the optimization in (10) does not over fit the observations. With L_c^* and d^*, our shape metric is,

$$\mu_s = f(\mathbf{L}_c^*, d^*). \tag{12}$$

It is worth noting that μ_s characterizes both the smoothness and the width requirement. Preferably μ_s should be small. In fact, d_l and d_r are the distances from the estimated boundaries to their respective observations, which means the area of the shaded area in Fig. 2 is μ_s. It is clear that μ_s becomes large if the lane markings do not correspond to a smooth desirable curve. The same applies to the lane that is overly wide or narrow. In such cases, μ_s becomes excessively large as shown in Fig. 2(b).

C. *Visibility Metric*

The visibility metric is defined based on both image pixels and LIDAR data. Define $\mu_{m,L}$ and $\mu_{b,L}$ as the mean intensity values for the lane marking points and the background points of the LIDAR scan, respectively. Recall that set $\mathcal{S}_Q$ contains lane marking pixels in image $\bar{\mathbf{I}}_t$, and define $\mathcal{S}_{b,I}$ to be the background pixel set. Define $\mu_{m,I}$ and $\mu_{b,I}$ to be the mean intensity values of the $\mathcal{S}_Q$ and $\mathcal{S}_{b,I}$ in $\bar{\mathbf{I}}_t$, respectively. The visibility metric is to verify intensity ratios in two modalities.

$$\mu_v = \max\left\{\frac{\mu_{m,L}}{\mu_{b,L}},\ \frac{\mu_{m,I}}{\mu_{b,I}}\right\}. \tag{13}$$

It is clear that large values of μ_v are preferable. As long as the lane markings are visible in either modality, we treat them as satisfactory here. It is also possible to change $\max$ to $\min$ if we want to be more conservative.

V. Dual Modal Lane Detection Algorithm

To compute the aforementioned metrics, we need the segmented left and right lane markings in both camera image and LIDAR data. This means that we need a lane detection algorithm. However, it is important to build this algorithm using the most common sensory configuration without catering to a particular hardware choice. In fact, it is not our best interest to use the best lane detection algorithm for LMQA purposes because we measure roads instead of vehicles. We need a baseline version of lane detection which can provide inputs required by our metrics. Unfortunately, existing commercial products only provide lane departure warnings instead of providing us with segmented pixels or

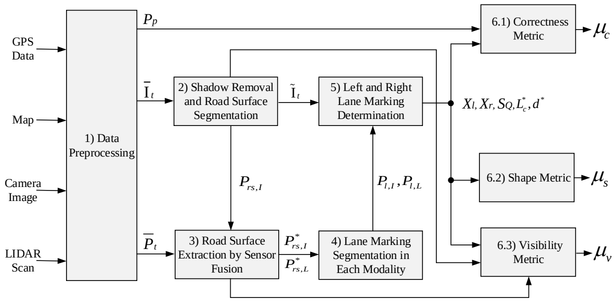


Fig. 3. Algorithm diagram.

coordinate. For completeness, we describe our lane detection algorithm here.

The overall sensor fusion pipeline is described in Fig. 3. In data preprocessing, we ensure all input data is synchronized. We then process camera images to remove shadows [27]. The shadow-free RGB image is recovered by relighting each pixel. For each image $\bar{\mathbf{I}}_t$, we extract the road surface by using fully convolutional networks [28], [29]. Note that we use full RGB colored images instead of gray-scale images in this step. Define $P_{rs,I} \subset \bar{\mathbf{I}}_t$ as road surface pixel set. Since noisy points are inevitable in $P_{rs,I}$, we need to fuse LIDAR data to reduce the influence of noise to refine the segmentation result.

A. Road Surface Extraction by Sensor Fusion

Define $\tilde{\mathbf{P}}_r = [x\ y\ z\ 1]^\intercal$ as the homogeneous 3D road surface point corresponding to the road surface image pixel $\mathbf{p}_r \in P_{rs,I}$. For each $\mathbf{p}_r$, we obtain the corresponding LIDAR point $\mathbf{P}_r$ using the inverse of (2),

$$\mathbf{P}_r = [\mathbf{KR}]^{-1}\tilde{\mathbf{p}}_r - \mathbf{R}^{-1}\mathbf{t}. \tag{14}$$

Assembling all $\mathbf{P}_r$, we obtain set $\mathcal{S}_r := \{\mathbf{P}_r\}$ containing the 3D LIDAR points belonging to road surface.

We model the road surface $S(x, y, z)$ using curved surface patches through bivariate polynomials [30],

$$z - \mathbf{H}_r \cdot \mathbf{z}_r = 0, \tag{15}$$

where $\mathbf{H}_r = [a_{00}\ a_{01}\ a_{11}\ a_{20}\ a_{21}\ a_{22}\ a_{30}\ a_{31}\ a_{32}\ a_{33}]^\intercal$ is the surface parameter vector that needs to be estimated, and $\mathbf{z}_r = [1\ x\ y\ x^2\ xy\ y^2\ x^3\ x^2y\ xy^2\ y^3]^\intercal$. We apply RANSAC [31] to filter out outliers and estimate the road surface model. A minimal solution can be established by randomly choosing 9 points from set $\mathcal{S}_r$ using a singular value decomposition (SVD) based algorithm. We set constraint $\|\mathbf{H}_r\|^2 = 1$ to avoid zero value solutions. In each iteration of RANSAC, we randomly select a minimal set of data from set $\mathcal{S}_r$ to estimate the $S(x, y, z)$. Denote $d_\perp(\mathbf{H}_r, \mathbf{P}_r)$ to be the shortest distance for a point $\mathbf{P}_r$ to the road surface, which is

$$d_\perp(\mathbf{H}_r, \mathbf{P}_r) = \min_{\mathbf{X}_r \in S} \|\mathbf{X}_r - \mathbf{P}_r\|, \tag{16}$$

subject to (15) for all $\mathbf{X}_r \in S(x, y, z)$, where $\mathbf{X}_r$ indicates a point on surface $S(x, y, z)$ that has the shortest distance to $\mathbf{P}_r$. By introducing the Lagrange multipliers λ, we find the point $\mathbf{X}_r$ on the road surface $\mathcal{S}$ for each $\mathbf{P}_r$ through solving the Lagrange function,

$$f_r(\mathbf{X}_r, \mathbf{P}_r, \mathbf{H}_r) = \|\mathbf{X}_r - \mathbf{P}_r\|^2 + \lambda(z - \mathbf{H}_r \cdot \mathbf{z}_r). \tag{17}$$

We employ the distance measurement in (16) to determine inlier/outlier from the set $\mathcal{S}_r$. Define $\mathcal{A}$ as the inlier consensus set of road surface. We accept an inlier point set if the ratio between the set cardinality of $\mathcal{A}$ and the sample size is greater than threshold τ_t. $\tau_t = 0.6$ in all experiments. After obtaining the largest consensus set, we refine the $\mathbf{H}_r$ using all inliers by applying the maximum likelihood estimation (MLE) to minimize the sum of distance errors,

$$\hat{\mathbf{H}}_r = \arg\min_{\hat{\mathbf{H}}_r} \sum_{\mathbf{P}_r \in \mathcal{A}} d_\perp(\mathbf{H}_r, \mathbf{P}_r)^2, \tag{18}$$

where $\hat{\mathbf{H}}_r$ denotes the estimation of $\mathbf{H}_r$ using the Levenberg-Marquardt (LM) algorithm.

After extracting road surface $S(x, y, z)$, we calculate the distance for all the LIDAR points in set $\{\overline{\mathcal{P}}_t \setminus \mathcal{A}\}$ using (16). We include the LIDAR points with distances to the surface less than threshold d_ϵ along with $\mathcal{A}$ itself,

$$\begin{aligned} P_{rs,L} = \big\{&\mathbf{P}_{i,t} | d_\perp(\hat{\mathbf{H}}_r, \mathbf{P}_{i,t}) \le d_\epsilon, \\ &\mathbf{P}_{i,t} \in \overline{\mathcal{P}}_t, \mathbf{P}_{i,t} \notin \mathcal{A}\big\} \cup \mathcal{A}. \end{aligned} \tag{19}$$

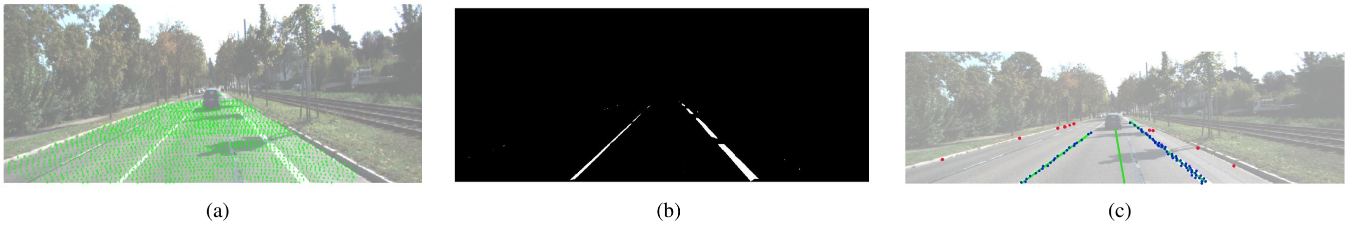
Define $\mathbf{p}_e$ to be the corresponding image pixel projection for the LIDAR point $\mathbf{P}_e \in P_{rs,L}$ on the image $\bar{\mathbf{I}}_t$ through (2). We adopt the DBSCAN clustering algorithm [32], [33] to eliminate outliers of pixel $\mathbf{p}_e$ located far away from the road surface. We then get the boundary and the interiors to obtain new road surface pixel set $P^*_{rs,I}$. We also remove the corresponding outliers from set $P_{rs,L}$ to get updated road surface LIDAR data set (see Fig. 4(a)). By employing the road surface model, we reduce the noise from image segmentation and include more LIDAR points that fit for the surface model, which reduces outliers for lane marking detection from the LIDAR scan in the later part.

B. Lane Marking Segmentation in Each Modality

With the road surface pixel $P^*_{rs,I}$ extracted, we detect lane markings from the segmented road surface in the image. Define $g(\mathbf{p}_{k,t})$ as the intensity value for k-th pixel $\mathbf{p}_{k,t} = [u\ v]^\intercal$.

We set the intensity value of non-road pixels in $\bar{\mathbf{I}}_t$ to be zero. Now $\bar{\mathbf{I}}_t$ only contains black pixels and road surface pixels including lane markings. Lane marking pixels usually have higher intensity values. To reduce the noise from the image, we apply Gaussian blurring before segmenting lane marking pixels through image histogram. We obtain the binned histogram according to 256 intensity levels. We apply Gaussian mixture model using EM algorithm [34] to the histogram data and find the peak with the largest intensity value μ_γ with variance σ_γ. By applying three-sigma thresholding [35], we obtain a lower bound of the intensity value as $g_\gamma = \mu_\gamma - 3\sigma_\gamma$. We obtain lane marking pixels $\mathbf{p}_l$ (see Fig. 4(b)) in set

$$P_{l,I} = \{\mathbf{p}_{k,t} | g(\mathbf{p}_{k,t}) \ge g_\gamma\}. \tag{20}$$



(a) (b) (c)

Fig. 4. Sample intermediate algorithm outputs. (a) The green points are the projected 3D LIDAR points from the road surface model by sensor fusion. (b) Lane marking pixels in image. (c) Blue points are the pixel-wise projection of the lane marking points from the LIDAR scan, red points are outliers, and the green curve in the middle is the lane center curve and the other two are the left and right lane boundaries, respectively (best viewed in color).

Fig. 4(b) illustrates the lane marking pixels.

Lane markings are also detected using LIDAR scans due to the their high laser reflectivity by design. Recall we have extracted road surface data from the LIDAR scan in set $P^*_{rs,L}$. Recall that variable $I_e \in [0, 255]$ to be the intensity value for LIDAR point $\mathbf{P}_e$. We threshold I_e to obtain lane markings in LIDAR data,

$$P_{l,L} = \{\mathbf{P}_e \mid I_e \geq T_s, \mathbf{P}_e \in P^*_{rs,L}\}, \qquad (21)$$

where threshold T_s is obtained using Otsu thresholding [36] that determines the optimal intensity value T_s by maximizing the variance between background (asphalt or concrete) and foreground (lane marking) classes.

C. Left and Right Lane Marking Determination

The lane markings from individual modalities can be further filtered through cross modality validation. Eqs. (14) and (2) allow us to project points between LIDAR coordinates and image coordinates back and forth. Hence, we can intersect the lane marking points between $P_{l,I}$ and $P_{l,L}$ at LIDAR coordinates and generate a set $P^*_{l,L}$ which contain dual-modal lane markings that are more robust than those in individual modalities.

At this moment, the lane markings may belong to several lane boundaries in a multi-lane highway and include many outliers. We first filter out all candidate lane boundaries before identifying the exact left and right lane boundaries. Define $\mathbf{L}_j$ as the j-the lane boundary. We apply T-Linkage [37] to obtain $\mathbf{L}_j$'s. T-Linkage is capable of detecting multiple lane boundaries in the presence of outliers but it requires a model for $\mathbf{L}_j$. We employ the cubic uniform B-spline lane boundary curve which is defined for a collection of $n+1$ control points $\mathcal{M}_l = \{\mathbf{P}_q\}$ from the set $P^*_{l,L}$ as,

$$\mathbf{L}_j(s) = \sum_{q=0}^{n} \mathbf{P}_q N_{q,3}(s). \qquad (22)$$

Here, $N_{q,3}(s)$ are the basis functions with

$$N_{q,0}(s) = \begin{cases} 1, & \text{if } s_q \leq s \leq s_{q+1} \text{ and } s_q < s_{q+1} \\ 0, & \text{otherwise} \end{cases}$$

$$N_{q,h}(s) = \frac{s - s_q}{s_{q+3} - s_q} N_{q,h-1}(s) + \frac{s_{q+4} - s}{s_{q+4} - s_{q+1}} N_{q+1,h-1}(s), \qquad (23)$$

where $h = 1, 2, 3$, $s_q = q - 3, q = 3, 4, ..., n+1$ with $s_0 = s_1 = s_2 = s_3$, and $s_{n+1} = s_{n+2} = s_{n+3} = s_{n+4}$. Note that the shape of the cubic uniform B-spline curve is dominated by the control points. We can impose curvature constraints when choosing points to instantiate models in T-Linkage.

After T-Linkage, we have a set of candidate lane boundaries $\mathbf{L}_j$'s. We need to identify left and right lane boundaries and their associated lane markings. A simple observation is that our left and right lane boundaries must intersect the low boundary of the image at positions closer to center of the low boundary because that is the current vehicle location. Recall that the horizontal dimension in the image is the u-axis. The intersection of $\mathbf{L}_j$ with low boundary generate u_j. If the center is at u_c, it is a natural divider for the left and right sides. Then we sort $|u_j - u_c|$ to generate two sorted sequences with increasing distances. We then pair them by considering the fact that the distance between left and right boundaries should be longer than $d_{\min}$ and shorter than $d_{\max}$. We might have multiple solutions but we use how close they are to the previous period to find the optimal. This simple search help us determine left and right boundaries, defined as $\mathbf{L}_l$ and $\mathbf{L}_r$, respectively. For each boundary, we find all closest points in $P^*_{l,L}$ and hence we determine $\mathcal{X}_l := \{\mathbf{X}_l\}$ and $\mathcal{X}_r := \{\mathbf{X}_r\}$ as the resulting left and right lane marking sets, respectively.

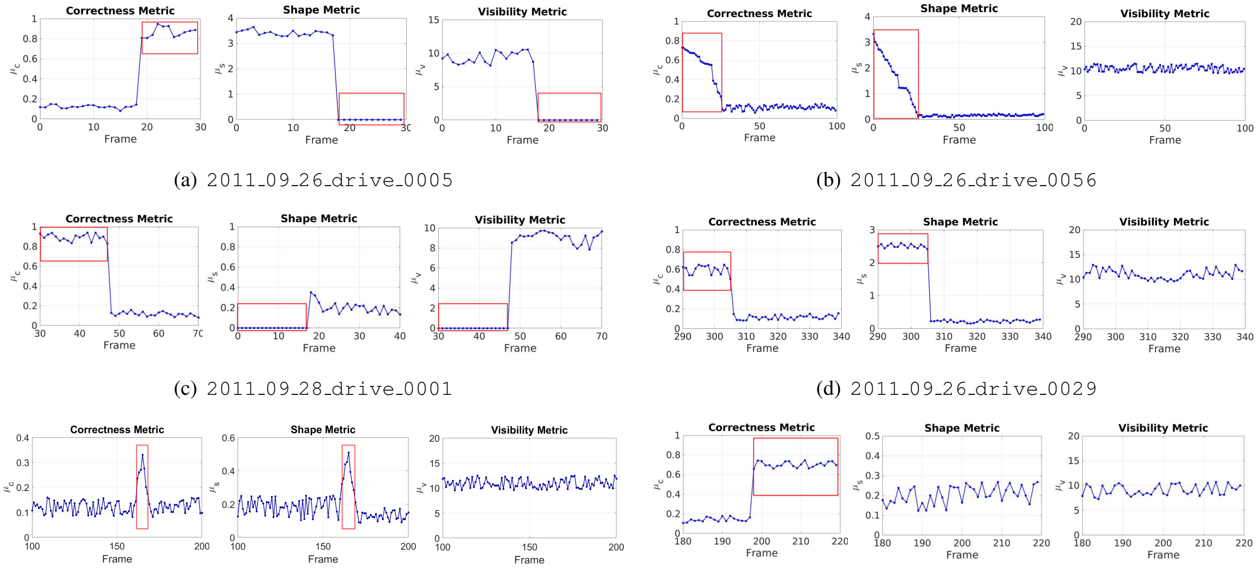
With the $\mathcal{X}_l$ and $\mathcal{X}_r$ obtained, we project them back to $\bar{\mathbf{I}}_t$ to search for more lane marking points. Denote $\mathbf{x}_{j,w}$ to be the corresponding projection pixel in $\bar{\mathbf{I}}_t$ for the LIDAR points in set $\mathcal{X}_l \bigcup \mathcal{X}_r$. We have lane marking pixel set

$$\mathcal{S}_Q = \{\mathbf{p}_{k,t} | \, \|\mathbf{p}_{k,t} - \mathbf{x}_{j,w}\| \leq d_j, \mathbf{p}_{k,t} \in \tilde{\mathbf{I}}_t, g(\mathbf{p}_{k,t}) \geq g_\gamma\}, \qquad (24)$$

where nonnegative variable d_j is a constant threshold value. Thus we have all values needed for metrics in Section IV.

VI. Experiments

We have implemented the proposed method on a Laptop PC with an Intel(R) Core$^{\text{TM}}$ i7-3517U CPU@1.90GHz and 8 GB memory. The Benchmark contains images showing a variety of street scenes captured from a vehicle driving around the city of Karlsruhe. Besides the raw data, KITTI comes with a number of labels for different tasks relevant to autonomous driving to evaluate the performance of our road extraction and lane detection results. Parameters are set according to the experiments empirically. We set $d_{\max}$ and



(a) 2011_09_26_drive_0005

(b) 2011_09_26_drive_0056

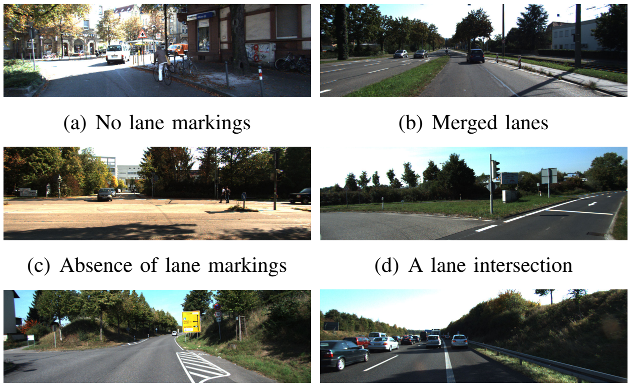
(c) 2011_09_28_drive_0001

(d) 2011_09_26_drive_0029

(e) 2011_09_26_drive_0070

(f) 2011_10_03_drive_0047

Fig. 5. Performance metrics for six sequences from KITTI dataset. Red boxes indicate lane marking anomalies identified by correctness, shape or visibility metrics.



(a) No lane markings

(b) Merged lanes

(c) Absence of lane markings

(d) A lane intersection

(e) A 3-way junction

(f) Right shoulder w/o markings

Fig. 6. Typical scenarios of abnormal lane markings. Figures labels correspond to those in Fig. 5.

$d_{\min}$ in (11) to be 4.60 m and 2.70 m, respectively. We also set d_ϵ in (19) to be 0.1 m, and d_j to be 20 pixels for (24).

To verify our metrics, we use six different sequences of two categories from KITTI dataset including city trail data (2011_09_26_drive_0005, 2011_09_26_drive_0056, 2011_09_28_drive_0001) and road trail data (2011_09_26_drive_0029, 2011_09_26_drive_0070, 2011_10_03_drive_0047). Fig. 5 illustrates testing results. Let us explain the abnormality of lane markings reflected by performance metrics as follows. Fig. 6(a) shows a case that lacks lane markings at the beginning of the video sequence. Fig. 6(b) shows a case that lanes start merging while the vehicle's current lane does not have left lane markings. Fig. 6(c) shows that the vehicle is entering a main road but the current lane does not have lane markings. Fig. 6(d) shows that an intersection does not have the lane markings to guide the vehicle. Fig. 6(e) shows a 3-way junction lacks part of the left lane markings and has irregular lane markings. Fig. 6(f) shows a case that one side of vehicle is just the shoulder with no right lane markings. To summarize, Fig. 5 shows that our metrics are able to capture the abnormality of the lane markings and can be used as a measurement tool for road inspection.

VII. Conclusion and Future Work

We focused on development of a LMQA method for improving infrastructure for autonomous driving. The method assumed an egocentric view from an inspection vehicle equipped with a GPS receiver, a frontal view camera, and a LIDAR for LMQA. We presented metrics and algorithms for lane marking assessment. Three lane marking quality metrics were proposed and modeled mathematically: correctness, shape, and visibility. We also proposed a dual-modal algorithm to facilitate the computation of the three metrics. We took both prior map uncertainty and sensory uncertainty into consideration in formulating our metrics and the algorithm. We implemented the algorithm and tested it under an open dataset. The results were satisfying. Our method was able to identify low quality segments of lane markings.

In the future, we will perform more tests and also consider other sensory combinations. We will also develop assessment algorithms for traffic signals, signs, and surface quality to form a complete inspection solution.

Acknowledgment

Thanks for C. Chou, H. Cheng, S. Yeh, A. Kingery, A. Angert, T. Sun, M. Jing, and Y. Yu for their inputs and contributions to the NetBot Laboratory, Texas A&M University.

References

[1] D. W. Harwood, A. D. May, I. B. Anderson, L. Leiman, and A. R. Archilla, "Capacity and quality of service of two-lane highways," *Final Report, NCHRP Project*, pp. 3–55, 1999.

[2] A. Flannery, K. Wochinger, and A. Martin, "Driver assessment of service quality on urban streets," *Transportation Research Record: Journal of the Transportation Research Board*, no. 1920, pp. 25–31, 2005.

[3] T. Veit, J.-P. Tarel, P. Nicolle, and P. Charbonnier, "Evaluation of road marking feature extraction," in *Intelligent Transportation Systems, 2008. ITSC 2008. 11th International IEEE Conference on*. IEEE, 2008, pp. 174–181.

[4] J. Pohl, W. Birk, and L. Westervall, "A driver-distraction-based lane-keeping assistance system," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 221, no. 4, pp. 541–552, 2007.

[5] K. Yamaguchi, A. Watanabe, T. Naito, and Y. Ninomiya, "Road region estimation using a sequence of monocular images," in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*. IEEE, 2008, pp. 1–4.

[6] V. Pradeep, G. Medioni, and J. Weiland, "Piecewise planar modeling for step detection using stereo vision," in *Workshop on computer vision applications for the visually impaired*, 2008.

[7] J. Hernández and B. Marcotegui, "Filtering of artifacts and pavement segmentation from mobile lidar data," in *ISPRS Workshop Laserscanning 2009*, 2009.

[8] J. Li, X. Mei, D. Prokhorov, and D. Tao, "Deep neural network for structural prediction and lane detection in traffic scene," *IEEE transactions on neural networks and learning systems*, vol. 28, no. 3, pp. 690–703, 2017.

[9] Y. Yu, J. Li, H. Guan, F. Jia, and C. Wang, "Learning hierarchical features for automated extraction of road markings from 3-d mobile lidar point clouds," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 8, no. 2, pp. 709–726, 2015.

[10] H. Guan, J. Li, Y. Yu, C. Wang, M. Chapman, and B. Yang, "Using mobile laser scanning data for automated extraction of road markings," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 87, pp. 93–107, 2014.

[11] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: survey, system, and evaluation," *IEEE transactions on intelligent transportation systems*, vol. 7, no. 1, pp. 20–37, 2006.

[12] A. B. Hillel, R. Lerner, D. Levi, and G. Raz, "Recent progress in road and lane detection: a survey," *Machine vision and applications*, vol. 25, no. 3, pp. 727–745, 2014.

[13] F. Samadzadegan, A. Sarafraz, and M. Tabibi, "Automatic lane detection in image sequences for vision-based navigation purposes," *ISPRS Image Engineering and Vision Metrology*, 2006.

[14] S. Kammel and B. Pitzer, "Lidar-based lane marker detection and mapping," in *Intelligent Vehicles Symposium, 2008 IEEE*. IEEE, 2008, pp. 1137–1142.

[15] A. von Reyher, A. Joos, and H. Winner, "A lidar-based approach for near range lane detection," in *Intelligent Vehicles Symposium, 2005. Proceedings. IEEE*. IEEE, 2005, pp. 147–152.

[16] K. Takagi, K. Morikawa, T. Ogawa, and M. Saburi, "Road environment recognition using on-vehicle lidar," in *Intelligent Vehicles Symposium, 2006 IEEE*. IEEE, 2006, pp. 120–125.

[17] L. T. Sach, K. Atsuta, K. Hamamoto, and S. Kondo, "A robust road profile estimation method for low texture stereo images," in *Image Processing (ICIP), 2009 16th IEEE International Conference on*. IEEE, 2009, pp. 4273–4276.

[18] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, D. Duggins, T. Galatali, C. Geyer *et al.*, "Autonomous driving in urban environments: Boss and the urban challenge," *Journal of Field Robotics*, vol. 25, no. 8, pp. 425–466, 2008.

[19] M. Montemerlo, J. Becker, S. Bhat, H. Dahlkamp, D. Dolgov, S. Ettinger, D. Haehnel, T. Hilden, G. Hoffmann, B. Huhnke *et al.*, "Junior: The stanford entry in the urban challenge," *Journal of field Robotics*, vol. 25, no. 9, pp. 569–597, 2008.

[20] U. Hofmann, A. Rieder, and E. D. Dickmanns, "Radar and vision data fusion for hybrid adaptive cruise control on highways," *Machine Vision and Applications*, vol. 14, no. 1, pp. 42–49, 2003.

[21] X. Gu, A. Zang, X. Huang, A. Tokuta, and X. Chen, "Fusion of color images and lidar data for lane classification," in *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems*. ACM, 2015, p. 69.

[22] A. S. Huang, D. Moore, M. Antone, E. Olson, and S. Teller, "Finding multiple lanes in urban road networks with vision and lidar," *Autonomous Robots*, vol. 26, no. 2-3, pp. 103–122, 2009.

[23] A. Mammeri, A. Boukerche, and Z. Tang, "A real-time lane marking localization, tracking and communication system," *Computer Communications*, vol. 73, pp. 132–143, 2016.

[24] B.-S. Shin, Z. Xu, and R. Klette, "Visual lane analysis and higher-order tasks: a concise review," *Machine vision and applications*, vol. 25, no. 6, pp. 1519–1547, 2014.

[25] S. P. Narote, P. N. Bhujbal, A. S. Narote, and D. M. Dhane, "A review of recent advances in lane detection and departure warning system," *Pattern Recognition*, vol. 73, pp. 216–234, 2018.

[26] C. E. Rasmussen, *Gaussian Processes for Machine Learning*. The MIT Press, 2006.

[27] R. Guo, Q. Dai, and D. Hoiem, "Single-image shadow detection and removal using paired regions," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011, pp. 2033–2040.

[28] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[29] E. Shelhamer, J. Long, and T. Darrell, "Fully convolutional networks for semantic segmentation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 4, pp. 640–651, 2017.

[30] R. Jain, R. Kasturi, and B. G. Schunck, *Machine vision*. McGraw-Hill New York, 1995, vol. 5.

[31] R. Raguram, J.-M. Frahm, and M. Pollefeys, "A comparative analysis of ransac techniques leading to adaptive real-time random sample consensus," *European Conference on Computer Vision*, pp. 500–513, 2008.

[32] D. Arlia and M. Coppola, "Experiments in parallel clustering with dbscan," in *European Conference on Parallel Processing*. Springer, 2001, pp. 326–331.

[33] H. B. Barua and S. Sarmah, "An extended density based clustering algorithm for large spatial 3d data using polyhedron approach," *International Journal of Computer Applications*, vol. 58, no. 2, 2012.

[34] N. M. Nasrabadi, "Pattern recognition and machine learning," *Journal of electronic imaging*, vol. 16, no. 4, p. 049901, 2007.

[35] J. M. Duncan, "Factors of safety and reliability in geotechnical engineering," *Journal of geotechnical and geoenvironmental engineering*, vol. 126, no. 4, pp. 307–316, 2000.

[36] M. H. J. Vala and A. Baxi, "A review on otsu image segmentation algorithm," *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, vol. 2, no. 2, pp. pp–387, 2013.

[37] L. Magri and A. Fusiello, "T-linkage: A continuous relaxation of j-linkage for multi-model fitting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 3954–3961.