

Illumination-Resilient Lane Detection by Threshold Self-Adjustment Using Newton-Based Extremum Seeking

Yujing Zhou^{ID}, Zejiang Wang^{ID}, *Graduate Student Member, IEEE*, and Junmin Wang^{ID}, *Senior Member, IEEE*

Abstract—The ability to detect lane markings under varying lighting conditions is essential for autonomous mobile robots and automatic vehicle driving assistance systems. Because the object color information is subject to illumination variation, this article presents a novel and computationally efficient algorithm based on the extremum-seeking method to achieve illumination-resilient lane detection and path-following tasks for autonomous driving. Lane detection is performed in the hue-saturation-value color space by distinguishing the colored lane marks from the background. The system's inputs are the upper and lower thresholds in each of the hue, saturation, and value channels. We define a cost function as the combination of detection accuracy and lane coverage to evaluate the algorithm performance. Two extremum-seeking schemes, one with fixed dither amplitudes and another with adaptive dither amplitudes, are designed to adjust the system inputs to minimize the cost function. The two proposed methods are validated and compared through video-based simulation studies and scaled-car field experimental tests.

Index Terms—Lane detection, automated driving, extremum seeking.

I. INTRODUCTION

LANE departure warning (LDW) is an essential feature for automated mobile robots and advanced driving assistance systems (ADAS) [1]. As a preliminary requirement, accurate lane detection is essential for the LDW system to work properly. Traditional image-processing-based lane detection consists of three processes: image pre-processing, edge detection, and lane modeling [2]. The color transformation from the red/green/blue (RGB) color space to grayscale is widely used during the image-preprocessing stage [3]. Then, edge detection methods extract line features based on the image intensity in grayscale, such as the Canny [4], Gabor [5], and Hough transform [6] algorithms. Finally, mathematical models are used to fit the extracted edges. Carefully chosen region of interest (ROI) [7] and smoothing operations with filters [8] are usually needed to remove unnecessary regions and falsely detected edges to achieve the desired results. However, they further increase the model complexity upon the already computationally heavy edge-detection algorithms [9].

Manuscript received 15 June 2021; revised 3 January 2022 and 15 March 2022; accepted 20 March 2022. Date of publication 4 April 2022; date of current version 11 October 2022. The Associate Editor for this article was Y. Wiseman. (Corresponding author: Junmin Wang.)

The authors are with the Walker Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX 78712 USA (e-mail: yz009@utexas.edu; wangzejiang@utexas.edu; jwang@austin.utexas.edu).

Digital Object Identifier 10.1109/TITS.2022.3161902

Apart from the traditional grayscale recognition methods, learning-based semantic segmentation methods [10], [11] have succeeded in event detection because of the increased representation and computational ability. Mainstream methods usually treat the lane detection problem as a pixelwise segmentation task, such as the VPGNet proposed by Lee *et al.* [12] and the self-attention distillation (SAD) algorithm proposed by Hou *et al.* [13]. However, they face problems with challenging scenarios and long runtime. To address these concerns, Qin *et al.* [14] proposed treating lane-detection as a row-based selection problem using global features, achieving a state-of-the-art runtime of 3.2 ms and accuracy of higher than 95%. Besides the segmentation formulation, clustering methods [15] are also popular in lane detection applications. Li *et al.* proposed a rank constrained spectral clustering (SC) to handle the issue of suboptimal learning [16] and an adaptive affinity graph for SC to enable multi-feature learning [17]. Despite the outstanding performance of learning-based methods, they have three inherent drawbacks. Firstly, mass labeled training datasets are not always available. Secondly, the training process is usually time-consuming, depending on the dataset and computing hardware. Finally, high-end GPU parallel computation is generally needed to reduce the computational time, e.g., NVIDIA GTX 1080Ti GPU was used in [14]. Such hardware requirement limits the application of the learning-based methods on mobile robots and scaled vehicles. Further, machine-learning based methods lack interpretability, which may be an important attribute for some applications.

Instead of using the grayscale gradient information from the traditional edge-detection algorithm, we propose a novel adaptive, learning-free, and computationally efficient lane detection method that leverages the color difference between lane markers and their background. It is computationally efficient because the lane detection task ends up being a comparison between two matrices to separate colors in lane markers and their background.

The problem of changing color appearances under varying lighting conditions has been a challenge for lane detection [18]–[20]. We address this problem by adaptively adjusting the thresholds in the HSV color space to maintain high lane detection accuracy under different lighting conditions. Using RGB color space has two major drawbacks: first, the correlated color information is sensitive to lighting variation; second, RGB color space does not explicitly consider the saturation

and brightness level [21]. Compared to the RGB color space, HSV color space separates the color, saturation, and brightness information into three channels. Any color can be specified by different thresholds on the H, S, and V channels with six thresholds in total: upper bound and lower bound in each of the three channels.

We propose using the extremum-seeking (ES) method to adjust the six thresholds simultaneously under different lighting conditions. ES [22]–[24] is a model-independent optimization routine, which can minimize a cost function of lane detection performance without knowing the explicit mathematical relationship between the optimized HSV thresholds and the cost function. This article presents two ES schemes: Newton-based ES with fixed and adaptive dither amplitudes.

The contributions of the proposed methods in this paper are as follows: 1) It can detect lane marks with high accuracy under varying lighting conditions without requiring a training process that is usually time-consuming; 2) The method is interpretable, which is an important attribute that some machine learning based methods cannot provide; 3) The proposed method is computationally efficient and can be implemented in real-time on mobile robots and scaled vehicles that cannot afford high-performance computational hardware.

The rest of this paper is organized as follows. Section II provides the problem formulation. Section III introduces two ES schemes. Simulation studies and field tests are shown in Sections IV and V, respectively. Finally, we conclude the paper in Section VI.

II. PROBLEM FORMULATION

This section elaborates on the problem of performing lane detection directly in the HSV color space and our illumination-resilient approach using the ES method.

Each color has a corresponding set of H, S, and V values in the HSV color space. Analogously, a range over each H, S, and V channel can specify multiple colors. Each range is defined by an upper and a lower threshold. It is preferred that these thresholds are automatically determined because the color shown in an image is subject to illumination variation.

The ES method adaptively changes the HSV color space thresholds by optimizing the system output under varying lighting conditions. We define the following objective function to evaluate the lane detection performance:

$$J = 1 - \phi\zeta, \quad (1)$$

where

$$\phi = \frac{\text{Number of identified pixels in the lane region}}{\text{Total number of identified pixels}}, \quad (2)$$

$$\zeta = \frac{\text{Number of identified pixels in the lane region}}{\text{Total number of pixels in the lane region}}. \quad (3)$$

In (1), ϕ is a measure of the identification accuracy ($\phi \in [0, 1]$), and ζ is a measure of the lane coverage ($\zeta \in [0, 1]$). With both ϕ and ζ being considered, the cost function simultaneously optimizes the identification accuracy and the total number of identified pixels in the lane region, which reduces the missing lane area. Note that the lane detection

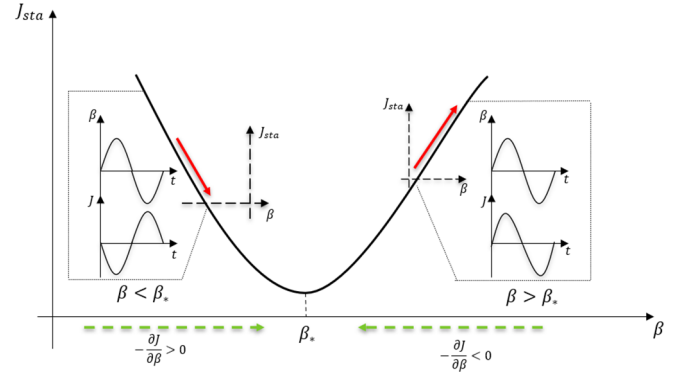


Fig. 1. Representation of the gradient estimation and time-scale separation behind the ES algorithm.

with ES methods does not require edge detection and lane modeling. However, the real-time lane location is needed *a priori* to evaluate the objective function J . This assumption can be achieved in reality, especially for industrial factory [25], warehouse [26], community services [27], and smart city [28] settings where intermittent stops, e.g., electric vehicle charging stations, are made during vehicle operation. We assume that the lighting conditions slowly vary between stops. The vehicle is provided with known lane locations and performs the ES adjustment while at rest, and it uses the adjusted thresholds during operation. The process of evaluating (1) is illustrated in detail in section IV.

III. EXTREMUM SEEKING METHODS

The goal of ES is to find the local optimal of an unknown mapping by adjusting the control variable following the gradient-descent direction. Since the function is unknown, the gradient information cannot be explicitly formulated. Instead, it is obtained via perturbing the control variables with sinusoidal dithers. This idea is shown in Fig. 1 [29]. In the plot, J and J_{sta} are the system cost functions from the perturbed input and under steady-state, respectively. β is the control variable. We assume that the cost function J_{sta} has a local minimum at β_* . When $\beta < \beta_*$, the dither signal $\sin(\omega t)$ will lead J to decrease. When $\beta > \beta_*$, the same dither signal will increase J . When β is in the vicinity of β_* , the dither signal will not cause J to vary by a significant amount. By following the negative gradient direction, which is shown by the dashed arrows in Fig. 1, the ES algorithm drives the control variable toward β_* . This procedure is used by the standard gradient-based ES. However, the estimated gradient depends on the unknown second-order derivative (Hessian). At different operating points, an increase in the Hessian can destabilize the system, and a decrease in the Hessian can lead to a slow convergence [30]. To resolve this issue, the Newton-based ES estimates the inverse of Hessian and makes the process Hessian-independent. This section presents two ES schemes: Newton-based ES with fixed and adaptive dither amplitude.

A. Newton-Based ES With Fixed-Amplitude Dither Signal

This section introduces the gradient-based ES with fixed dither amplitude, then extends it to the Newton-based version.

The overall structure of a Newton-based ES with fixed and adaptive dither amplitudes is shown in Fig. 2. The gradient-based ES has a simpler version of Fig. 2 without using the enclosed sections labeled as “Inverse of Hessian Estimation” and “Adaptive Dither Amplitude.” The ES simultaneously controls six parameters: the lower and upper thresholds in each of the H, S, and V channels. Thus, there are six dither signals in $S(t)$ as defined below:

$$S(t) = [a_1 \sin(\omega_1 t) \quad \cdots \quad a_6 \sin(\omega_6 t)], \quad (4)$$

where a is the dither amplitude, and ω is the dither frequency. Without using the adaptive dither scheme, the online-tuned dither amplitude \hat{a} equals 1, which preserves the fixed value of $S(t)$ in (4). As shown in Fig. 2, the system input is denoted as β . With perturbed system inputs $\beta = \beta_0 + S(t)$, the system output from the objective function J is expressed using the Taylor series expansion as follows:

$$\begin{aligned} J(\beta) &= J(\beta_0 + S(t)) \\ &= J(\beta_0) + \sum_{i=1}^n [a_i \sin(\omega_i t) \nabla J_i] + O(a_i^2 \sin^2(\omega_i t)). \end{aligned} \quad (5)$$

The gradient information ∇J is contained within the AC component of (5). Therefore, a high-pass filter $\frac{s}{s+\omega_h}$ is used to filter out the DC component, and the higher-order component in (5) is neglected. The filtered cost function $\eta(t)$ is expressed as:

$$\eta(t) = \frac{s}{s+\omega_h} J(\beta) = \sum_{i=1}^n [a_i \sin(\omega_i t) \nabla J_i]. \quad (6)$$

To separate ∇J in (6), which corresponds to the “Gradient Estimation” process shown in Fig. 2, we multiply (6) by the same dither signals in (4) and obtain:

$$\begin{aligned} M_i &= S(t) \sum_{i=1}^n [a_i \sin(\omega_i t) \nabla J_i] \\ &= \frac{a_i^2}{2} (1 - \cos^2(2\omega_i t)) \nabla J_i \\ &\quad + \sum_{m=1}^{n-1} \sum_{k=k+1}^n a_m a_k \sin(\omega_m t) \sin(\omega_k t) \nabla J_k. \end{aligned} \quad (7)$$

Eq. (7) shows that the gradient information is carried within the DC component of M_i . Thus, AC terms in (7) are filtered out by the low-pass filter $\frac{\omega_l}{s+\omega_l}$. Then, (7) is simplified as:

$$\hat{G}_i = \frac{\omega_l}{s+\omega_l} M_i = \frac{a_i^2}{2} \nabla J_i. \quad (8)$$

With the estimated gradient in (8), the system input β is updated following the negative gradient direction with control gain $-K$, as shown in the “Optimization” stage in Fig. 2. The process is repeated until the control variable is in the vicinity of β_* and the cost function J converges to its local extremum, as shown in Fig. 1. However, this method has a problem that motivates the development of the Newton-based ES.

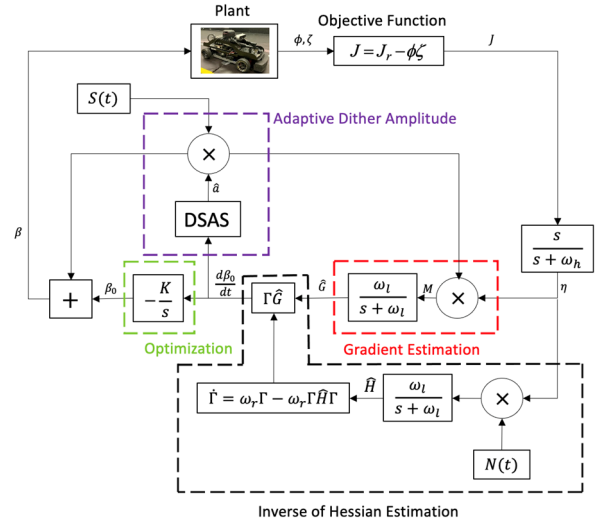


Fig. 2. Schematic of a multivariable Newton-based ES with fixed-dither and adaptive-dither.

Note that the optimal control variable is assumed to be β_* . Using Taylor series expansion, we can express the gradient of the cost function J at β_* as:

$$\nabla J = \frac{\partial^2 J}{\partial \beta^2} \Big|_{\beta=\beta_*} (\beta_0 - \beta_*) + O((\beta_0 - \beta_*)^2). \quad (9)$$

Higher-order terms in (9) are neglected. It shows that the system convergence rate is governed by the second-order derivative (Hessian), which we do not know from the gradient estimation processes in (5)-(8). The Newton-based ES solves this problem by estimating the Hessian of the cost function and multiplying the estimated gradient by the inverse of Hessian to make the process Hessian-independent. This process is illustrated in the “Inverse of Hessian Estimation” part in Fig. 2. Here, we also assumed fixed dither amplitudes with $\hat{a} = 1$. To explain the process of Hessian estimation, we include the second-order term in (5) and rewrite (6) as:

$$\begin{aligned} \eta(t) &\approx S(t)^T \nabla J + \frac{1}{2} S(t)^T H S(t) \\ &= \sum_{i=1}^n a_i \sin(\omega_i t) \nabla J_i \\ &\quad + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n a_i a_j \sin(\omega_i t) \sin(\omega_j t) H_{ij}, \end{aligned} \quad (10)$$

where H is the Hessian. In Fig. 2, \hat{H} and Γ are the estimation of Hessian and the inverse of Hessian, respectively. We follow the guidance in [31] and select the following elements for $N(t)$:

$$N_{ii} = \frac{16}{a_i^2} (\sin^2(\omega_i t) - \frac{1}{2}), \quad (11)$$

$$N_{ij} = \frac{4}{a_i a_j} \sin(\omega_i t) \sin(\omega_j t), \quad (12)$$

where $ij = 1, \dots, 6$ and $i \neq j$. Multiplying $\eta(t)$ in (10) with $N(t)$ yields:

$$\begin{aligned} \eta N_{ii} &= \frac{16}{a_i^2} (\sin^2(\omega_i t) - \frac{1}{2}) \sum_{i=1}^n a_i \sin(\omega_i t) \nabla J_i \\ &\quad + \frac{8}{a_i^2} (\sin^2(\omega_i t) - \frac{1}{2}) \sum_{i=1}^n \sum_{j=1}^n a_i a_j \sin(\omega_i t) \\ &\quad \times \sin(\omega_j t) H_{ij}, \end{aligned} \quad (13)$$

$$\begin{aligned} \eta N_{ij} &= \frac{4}{a_i a_j} \sin(\omega_i t) \sin(\omega_j t) \sum_{i=1}^n a_i \sin(\omega_i t) \nabla J_i \\ &\quad + \frac{2}{a_i a_j} \sin(\omega_i t) \sin(\omega_j t) \sum_{i=1}^n \sum_{j=1}^n a_i a_j \sin(\omega_i t) \\ &\quad \times \sin(\omega_j t) H_{ij}. \end{aligned} \quad (14)$$

H is symmetric with diagonal terms H_{mm} and cross-terms H_{mk} , where $m \neq k$. We separate the terms H_{mm} and H_{mk} from (13) and (14) and obtain the following expressions:

$$\begin{aligned} \eta N_{mm} &= \frac{16}{a_i^2} (\sin^2(\omega_i t) - \frac{1}{2}) \sum_{i=1}^n a_i \sin(\omega_i t) \nabla J_i \\ &\quad + \frac{8}{a_i^2} (\sin^2(\omega_i t) - \frac{1}{2}) \sum_{i=1, i \neq m}^n \sum_{j=1, j \neq m}^n a_i a_j \sin(\omega_i t) \\ &\quad \times \sin(\omega_j t) H_{ij} + 4[1 - 2\cos^2(2\omega_m t) \\ &\quad + \cos(2\omega_m t)] H_{mm}, \end{aligned} \quad (15)$$

$$\begin{aligned} \eta N_{mk} &= \frac{4}{a_i a_j} \sin(\omega_i t) \sin(\omega_j t) \sum_{i=1}^n a_i \sin(\omega_i t) \nabla J_i \\ &\quad + \frac{2}{a_i a_j} \sin(\omega_i t) \sin(\omega_j t) \sum_{i=1, i \neq m}^n \sum_{j=1, j \neq k}^n a_i a_j \sin(\omega_i t) \\ &\quad \times \sin(\omega_j t) H_{ij} + [1 - \cos(2\omega_m t) - \cos(2\omega_k t) \\ &\quad + \frac{1}{2} \cos(2\omega_m t \pm 2\omega_k t)] H_{mk}. \end{aligned} \quad (16)$$

As shown in (15) and (16), the Hessian information is carried within the DC component of ηN . Hence, we use a low-pass filter to eliminate the AC components and obtain the estimated Hessian, \hat{H} .

A direct inverse of the Hessian causes problems when \hat{H} is close to singular. So, a dynamic estimator is used to estimate the inverse of \hat{H} by solving the following Riccati equation:

$$\dot{\Gamma} = \omega_r \Gamma - \omega_r \Gamma \hat{H} \Gamma, \quad (17)$$

where Γ is the estimated inverse of \hat{H} . Equation (17) exponentially converges to $\Gamma_* = \hat{H}^{-1}$, and the proof is shown in [31]. Finally, the effect of Hessian on the gradient estimation can be eliminated by multiplying it with Γ , which concludes the Newton-based ES algorithm.

B. Newton-Based Extremum Seeking With Adaptive Dither Amplitude

According to the proof of stabilities for both the gradient-based and the Newton-based ES from Krstić and Wang [32] and Ghaffari *et al.* [31], sufficiently small dither amplitude

a , control gain K , and dither frequency ω must be chosen to achieve time-scale separation among the plant, the perturbation, and the optimization process. Only if this time-scale separation is achieved, can the measured system behavior J be close to the steady-state performance map J_{sta} . With appropriately chosen initial conditions close to the local extremum, the ES algorithm can optimize the unknown mapping J and the control variable β to a $O(\omega + \delta + a)$ -neighborhood of their local extrema.

As shown from (8), the system convergence speed is governed by both the optimization gain K and the dither amplitude a . Therefore, we prefer relatively large control gains and dither amplitudes to increase the optimizing rate. On the other hand, small dither amplitudes can drive control variables and system output to a smaller region around the local extremum. To balance the tradeoff between the convergence speed and the steady-state residual, the dither amplitude ought to be large when the system is far from its local optimal point and small when it is close to the extremum. Moase *et al.* [30] proposed a dither signal amplitude schedule (DSAS) to adjust the dither amplitude proportionally to the rate of change of control variables. Compared to the fixed-dither ES shown in Section III-A, the adaptive dither amplitude enables a faster convergence speed with an initially high dither amplitude when the objective function is far from the local optimal and less oscillation near convergence with a small dither amplitude when the objective function is close to the local optimal.

We integrate the Newton-based method with the DSAS proposed by Moase *et al.* [30] as shown in Fig. 2. The DSAS is designed such that the dither amplitude is large when the changing rate of β_0 is large and small when it is small. Therefore, the input to the DSAS is $\frac{d\beta_0}{dt}$, and its output is the adaptive dither amplitude \hat{a} . We use the same structure as presented in [30] with the following rule:

$$\frac{d\hat{a}}{dt} = K_a K \omega (\alpha - \hat{a}), \quad (18)$$

where

$$\alpha = \max\left(\left|\frac{\gamma}{\omega K} \frac{d\beta_0}{dt}\right|, a_{min}\right). \quad (19)$$

In (19), K_a is the amplitude gain, γ is a manually selected constant, and a_{min} is the minimum dither amplitude. Each variable in (18) and (19) is in \mathbb{R}^6 for our application.

IV. SIMULINK-QUARC JOINT SIMULATION

To validate and compare the two proposed ES methods, we first conducted Simulink-QUARC joint simulation studies. QUARC is a real-time control software that can interact with a scaled car by sending and collecting real-time data.

A. Simulation Setup

The simulation setup is shown in Fig. 3 where Fig. 3 (a) shows the block diagram for the lane detection process, and Fig. 3 (b) shows the scaled car with critical components labeled. For simulation purposes, we recorded videos from the front camera and applied them offline to the simulation. Each image from a video frame was made of a three-layer

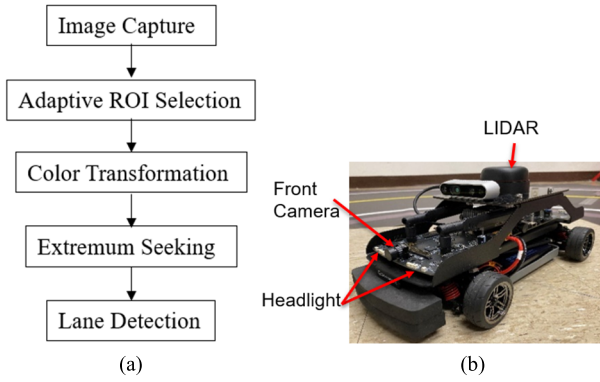


Fig. 3. (a) Simulation setup block diagram and (b) system hardware.

matrix, and the bottom half and left (right) one-third of the matrix were selected to be the ROI for the white lane (yellow lane) detection. Each image's selected ROI was transformed from RGB space to HSV space. Six thresholds, including the upper and lower bounds for each of the H, S, and V channels, were applied to the image to separate the lane markers from their backgrounds.

For simulation purposes, the scaled car stayed still with the lane location fixed in the image captured by its front camera. The lane marker's location was manually calibrated. With the available ground-truth lane location, metrics (2) and (3) can be evaluated from the ES identification results. Initial conditions for ES are critical to the algorithm's performance [31], [32]. Selected initial thresholds in the HSV color space and tuned controller parameters are tabulated in Table I. Three lighting sources in an indoor environment were used to test the proposed algorithm: fluorescent ceiling light fixture (bright), LED light stand (medium), and the scaled car's headlight (dark). Under each lighting condition, white and yellow lane detections were performed.

To demonstrate that indoor lighting conditions can mimic the outdoor conditions, we compared 100 frame shots from the indoor bright and dark conditions with the outdoor counterparts in the HSV color space, and the results are shown in Fig. 4. Pixels within the white lane and background regions are plotted on the HSV-coordinate system. It can be seen from Fig. 4 that red and orange points are clustered together since they represent the pixels within the lane region, and blue and cyan points are clustered together since they represent the pixels within the background region. It infers that the critical information in lane detection tasks from an indoor image is close to the outdoor image. This analysis is by no means a technical proof that the indoor lighting condition can represent the complex outdoor environment, but it demonstrates that the dataset we used in this study is not limited to the indoor condition.

B. Simulation Results

The cost functions for white and yellow lane detections under bright, medium, and dark conditions are shown in Fig. 5. In each plot, the fixed-dither and adaptive-dither ES results are shown in solid and dashed lines, respectively. Before the ES

 TABLE I
PARAMETERS FOR THE ES ALGORITHMS

Symbol	DEFINITION	Value	
		Yellow Lane	White Lane
H_{\min}	Lower threshold initial condition for H channel	0	0
H_{\max}	Upper threshold initial condition for H channel	255	100
S_{\min}	Lower threshold initial condition for S channel	0	100
S_{\max}	Upper threshold initial condition for S channel	50	255
V_{\min}	Lower threshold initial condition for V channel	150	150
V_{\max}	Upper threshold initial condition for V channel	255	255
a	Dither amplitude	$0.5 \times [2; 3; 2; 3; 5; 2]$	
ω	Dither frequency	$[0.088; 0.432; 0.184; 0.296; 0.656; 0.328]$	
K	Control gain	$5 \times [30; 20; 20; 10; 15; 20]$	
ω_f	Hessian inverse estimation filter frequency	0.01	
Γ_0	Initial conditions for Hessian inverse estimation	$0.5 \times eye(6)$	
K_a	Dither amplitude control gain	$0.001 \times [1; 1; 1; 1; 1; 1]$	
a_{\min}	Minimum dither amplitude	$0.01 \times [2; 3; 2; 3; 5; 2]$	
γ	Constant to determine the adaptive dither amplitude	0.5	

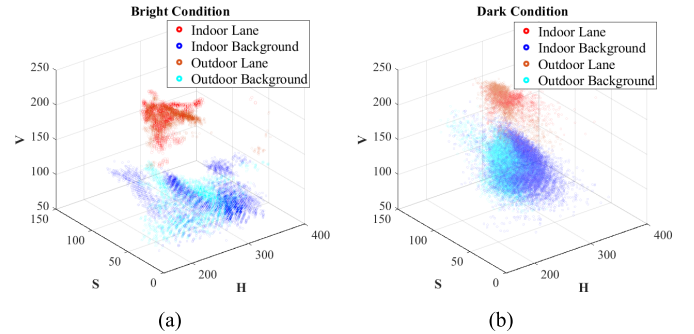


Fig. 4. Comparison of images from indoor and outdoor environments under (a) bright and (b) dark conditions. Red and orange clusters are close, and blue and cyan clusters are close, showing that the indoor condition is close to the outdoor one.

algorithm was activated, the system ran with default thresholds for 15 seconds, which allowed the cost function J to reach a steady state. The costs are high with the default thresholds, which is unsatisfactory for lane detection. With the ES being activated, the system output converges to less than 0.1 within 25 seconds (excluding the initial 15 seconds) in all scenarios. We can see from Fig. 5 (a)-(d) that the results from the adaptive-dither ES have less oscillation after convergence. This observation is consistent with [31] and [32]. Smaller dither amplitudes from the adaptive ES allow the cost function to converge closer to the local extremum.

The difference between two ES schemes becomes more pronounced if we look at the behavior of each control variable

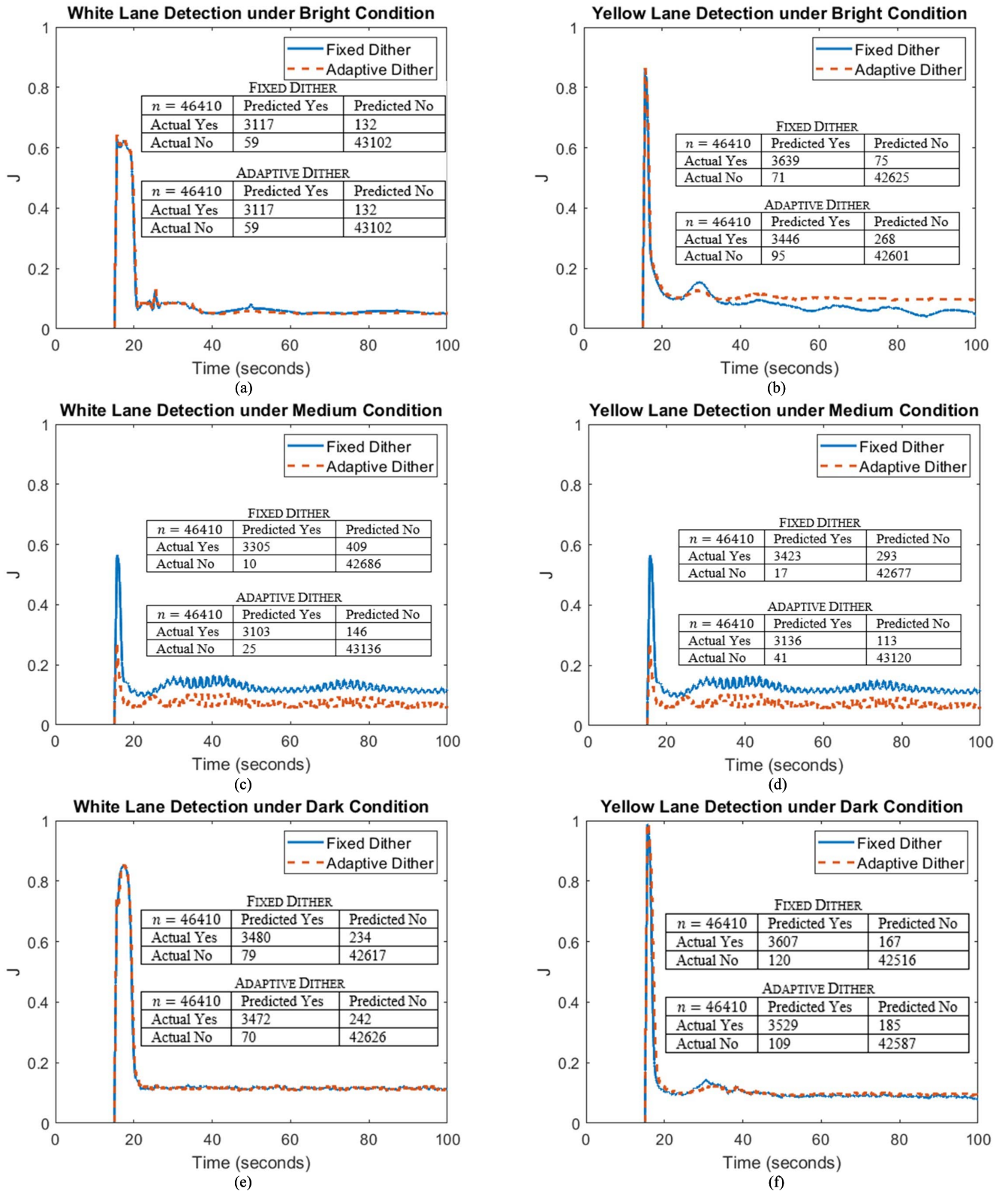


Fig. 5. Cost functions from simulation studies for white and yellow lane detections under bright, medium, and dark lighting conditions. Confusion matrices are included for each condition. The number of pixels that fall into the false positive and false negative categories is less than 1% of the total number of pixels in an image, respectively. The identification accuracy is over 99%.

shown in Fig. 6. We only present results under the bright and dark conditions here for simplicity. Every H, S, and V threshold is limited to a default range of 0 – 255. The scale in the H channel is changed from 0 – 360 to 0 – 255. In the

TABLE II
DITHER AMPLITUDES FOR FIXED DITHER ES AND ADAPTIVE DITHER ES

		White Lane Detection						Yellow Lane Detection					
		H_{\min}	S_{\min}	V_{\min}	H_{\max}	S_{\max}	V_{\max}	H_{\min}	S_{\min}	V_{\min}	H_{\max}	S_{\max}	V_{\max}
Fixed Dither		1	1.5	1	1.5	2.5	1	1	1.5	1	1.5	2.5	1
Adaptive Dither	Bright	0.178	0.033	0.143	0.206	0.052	0.029	0.175	0.033	0.143	0.204	0.052	0.029
	Dark	0.175	0.033	0.142	0.206	0.052	0.029	0.172	0.033	0.142	0.204	0.052	0.029

following analysis, we use $[min, max]$ to represent the lower and upper thresholds in a channel. For the white lane detection, the thresholds in the H channel remain at their initial conditions since the white color is independent of the H channel but has low S values and high V values. As expected, thresholds in the V channel converge to $[180, 255]$. Thresholds in the S channel converge to $[0, 255]$ from the fixed-dither ES but to $[0, 155]$ from the adaptive-dither ES. The difference comes from the dither amplitude self-adjustment for the adaptive-dither ES. Table II shows that dither amplitudes under the adaptive-dither ES are much smaller than that of the fixed dither ES after convergence. The smaller dither amplitude from the adaptive-dither ES allows the upper threshold in the S channel to increase at a lower rate than the fixed-dither ES and converges to a different local extremum in Fig. 6. Furthermore, it is worth noting that most of the background region is black, and the thresholds in the V channel play a more critical role in filtering out dark colors. In other words, different saturation values correspond to the same identification result since the V channel has filtered out the background color. From the optimization perspective, the cost function J is “flat” with respect to the S channel in identifying the white lane. Therefore, ES can generate different upper bounds for the S channel with the same cost. Interestingly, the converged thresholds are similar for both lighting conditions. Since the scaled car has white headlights, we conjecture that white lane markers look similar in an image under both lighting conditions.

For the yellow lane detection, thresholds in the H channel are important, and they converge to $[0, 130]$. Yellow is within this range in the HSV color space. Thresholds are higher in the V channel under the bright lighting condition because the V channel measures the brightness of a color.

Although cost functions shown in Fig. 5 do not converge to zero, this observation is expected. The ES method only guarantees a local extremum. We use confusion matrices to validate the performance of each ES method quantitatively. They are shown together with the cost function in Fig. 5. In each confusion matrix, n is the total number of pixels in an image, and four classes of information are included:

- 1) True positive (TP): predicted yes and actual yes,
- 2) True negative (TN): predicted no and actual no,
- 3) False positive (FP): predicted yes and actual no,
- 4) False negative (FN): predicted no and actual yes.

They can be related to ϕ and ζ in (2) and (3) as follows:

$$\phi = \frac{TP}{TP + FP}, \quad (20)$$

$$\zeta = \frac{TP}{TP + FN}. \quad (21)$$

The identification accuracy is defined as:

$$\frac{TP + TN}{TP + FP + TN + FN}. \quad (22)$$

For both ES methods, pixels within the FP and FN categories are less than 1% of the total number of pixels in an image. The identification accuracy is higher than 99%. It indicates that the lane identification task under varying lighting conditions performs well with both ES methods.

As a comparison to the state-of-the-art deep-learning result for lane detection, the trained network from Qin *et al.* [14] only attains an accuracy of 70.5% on our collected image data, which is significantly lower than the accuracy of 99% from the proposed ES methods. It indicates that re-training for different testing scenarios is required to generate good performance for the learning-based method. Without changing the network model, we trained it on our collected image data, and the testing result is 95.5%, which is still lower than the proposed ES methods. Note that the accuracy metric in (22) is different from the F1-measure used in [14]. For a fair comparison, we obtained the F1-measure accuracy of 96.3% and 96.4% for the fixed- and adaptive-dither ES, respectively, which are higher than the trained network model. Moreover, the training times for the Tusimple and the self-collected datasets on our local machine are respectively more than 5 hours and 1 hour, which are much longer than the convergence time of the proposed ES methods (less than 25 seconds).

The method in [14] reported a runtime of 3.2 *ms*, which was the time to detect all the lane markers in a single image frame using a trained network. By analogy, assuming that the thresholds in the HSV color space were obtained using the proposed ES scheme, the lane detection task in our proposed algorithm is an element-wise comparison of two 3-D matrices, one for the original image data and the other for the pre-determined thresholds, and its runtime is less than 1 *ms*.

V. SCALED CAR FIELD TESTS

The ES algorithms are further investigated on a scaled, automated driving car by letting it follow an indoor map.

A. Experimental Setup

The overall experimental setup is shown in Fig. 7. We used a camera-based indoor GPS to evaluate the scaled car's lane following performance. The system had eight infrared cameras that take multiple 2D images to reconstruct the 3D coordinates of the scaled car. The synchronized information from the eight

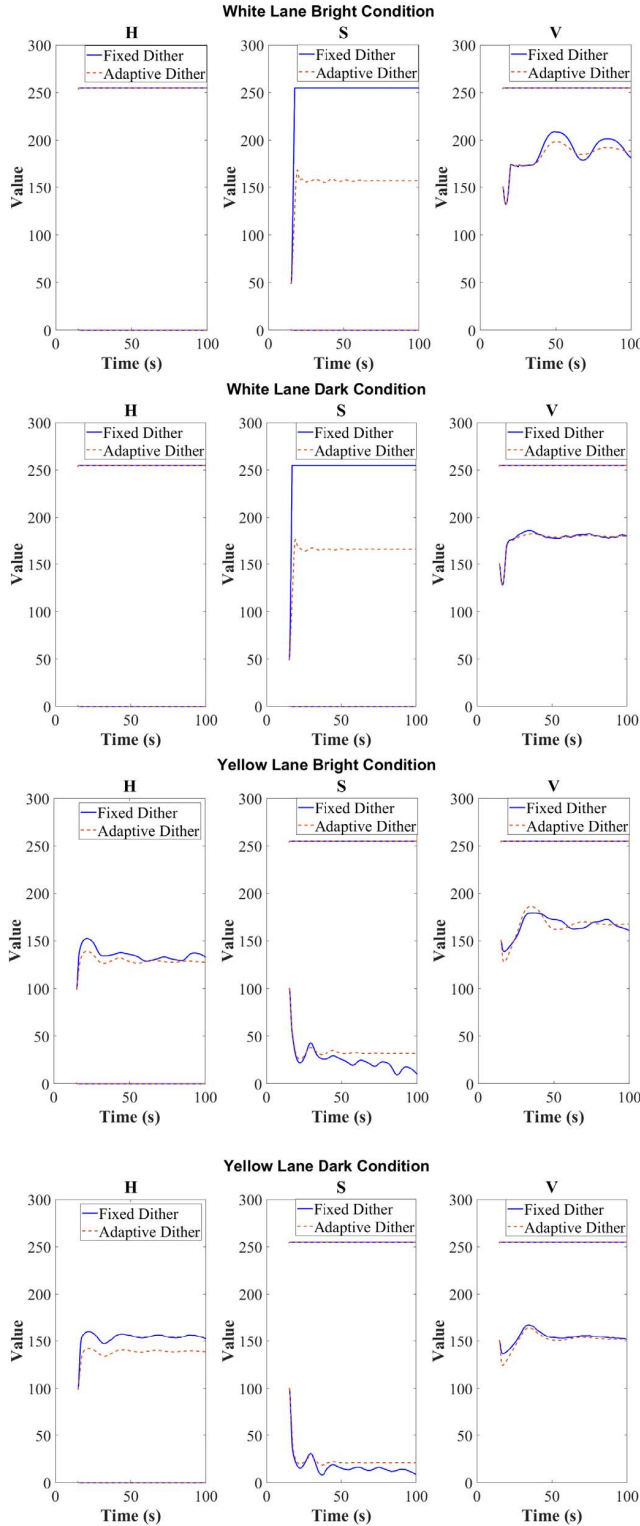


Fig. 6. Control variables from each ES method. each variable converges to similar points from the two ES methods except for the upper bound in the S channel for the white lane identification under bright conditions, which shows that the two ES methods drive the system to different extrema. It can be observed that thresholds from the adaptive dither ES are more stable.

cameras yields the vehicle's real-time location. Compared to the parameters listed in Table I, the control gain K was reduced to $[30; 20; 20; 10; 15; 20]$, and the amplitude gain K_a

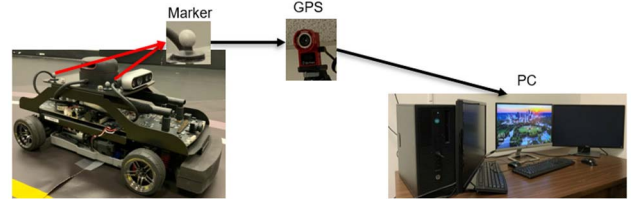


Fig. 7. Field test setup. The scaled car's real-time location is collected from an inoor-GPS system.

was increased to $0.01 \times [1; 1; 1; 1; 1; 1]$ for the experiment. Other parameters remained the same.

A steering control algorithm based on the linear polynomial fit of detected pixels within lane regions was used to test the performance of lane detection and following. We used an indoor map with white and yellow lane marks for the lane-following test, and it consisted of straight and circular portions. The scaled car ran the ES algorithm at the straight part of the map and used the adjusted H, S, and V thresholds to follow the circular lanes. The same scaled car and lane detection processes shown in Fig. 3 were used here. The difference between this field test and the simulation studies was that the actual lane location needed to be determined in real-time.

The distance between the straight lane markers and road infrastructure was assumed to be fixed and known when the scaled car stayed still and performed the ES algorithm. We chose the wall as infrastructure in the field test, and the white and yellow lanes were fixed at 0.98 m and 0.61 m away from it, respectively. Due to the image convergence effect, images were transformed to the top view, which preserved the road geometric topology information, such as the parallelism of two lanes [33], [34]. We offline calibrated the relationship between the Lidar measurement and the lane location. The result for the white lane is:

$$LB = 70 + 729.48(x - 0.82), \quad (23)$$

$$RB = LB + 20, \quad (24)$$

and the result for the yellow lane is:

$$LB = 148 + 972.64(x - 0.82), \quad (25)$$

$$RB = LB + 25, \quad (26)$$

where LB and RB are respectively the left and right edges of a lane marker in terms of pixels counted from the left edge of an image, and x is the real-time distance between the scaled car and the wall measured by the Lidar in meters.

B. Experimental Protocol

The experiment was designed to follow the procedure below for both white and yellow lane identifications:

- 1) Turn on the roof light. The scaled car follows the indoor map with default thresholds shown in Table I for 1 minute.
- 2) The scaled car stops for 5 minutes and performs the threshold adjustment using the fixed dither ES algorithm.
- 3) The scaled car follows the indoor map with the adjusted thresholds for 1 minute.

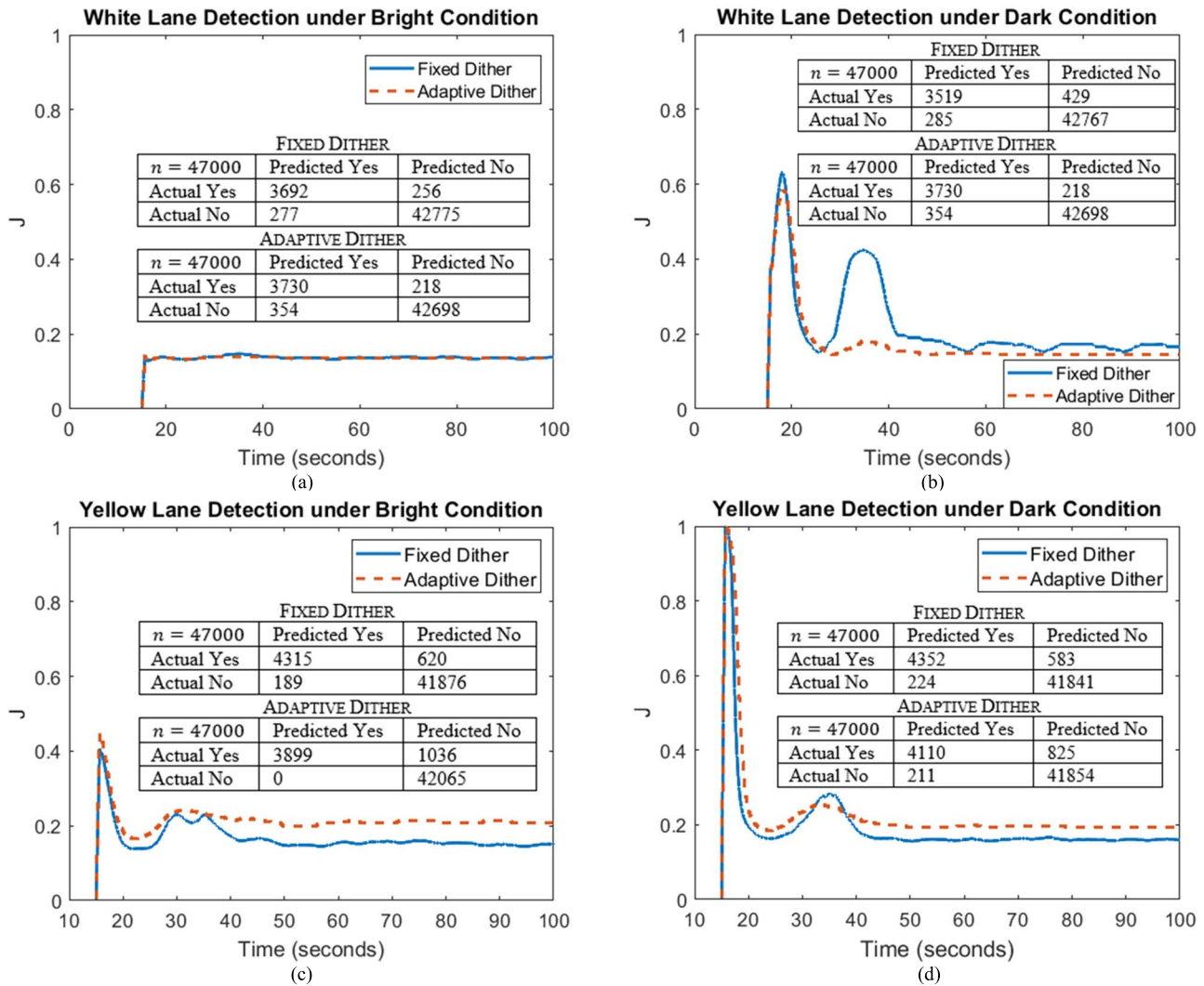


Fig. 8. Cost functions from the field test for white and yellow lane detections under bright and dark lighting conditions. Confusion matrices are included for each condition. The number of pixels that fall into the false positive and false negative pixels is less than 0.75% and 2.2% of the total pixels, respectively. The identification accuracy is over 97.8%.

- 4) The scaled car stops for 5 minutes and performs the threshold adjustment using the adaptive dither ES algorithm.
- 5) The scaled car follows the indoor map with the adjusted thresholds for 1 minute.
- 6) Turn off the roof light. The scaled car follows the indoor map for another 1 minute.
- 7) Repeat steps 2 to 5 to run both ES schemes under the dark condition.

C. Experimental Results

The lane identification results are shown in Fig. 8. In each plot, the fixed dither and adaptive dither ES results are shown in solid lines and dashed lines, respectively. The field test environment is slightly different from the simulation setup because of the painted floor. Hence, the cost function values shown in Fig. 8 are different from Fig. 5. In Fig. 8 (a), the starting point and converged values are close to each

other, which means that the initial threshold guesses are accurate. Lane detection improvements are clearly shown in Fig. 8 (b-d). There is less oscillation for converged values from the adaptive dither ES than that of the fixed dither ES as shown in Fig. 8 (a) and (b). Confusion matrices in each plot provide quantitative insight into the performance of ES schemes. For both methods, the pixels that fall into the FP and FN categories are less than 0.75% and 2.2% of the total number of pixels in an image, respectively. The identification accuracy is higher than 97.8%. The result is slightly worse than the simulation outcome, but this is expected since the lane locations estimated from Lidar measurements are less accurate than the manual calibration.

For simplicity, plots for control variables are not shown here. Instead, the values for the converged H , S , and V thresholds are listed in Table III. Thresholds in the H channel are $[0, 255]$ for the white lane identification because the white color is independent of the H channel in the HSV space.

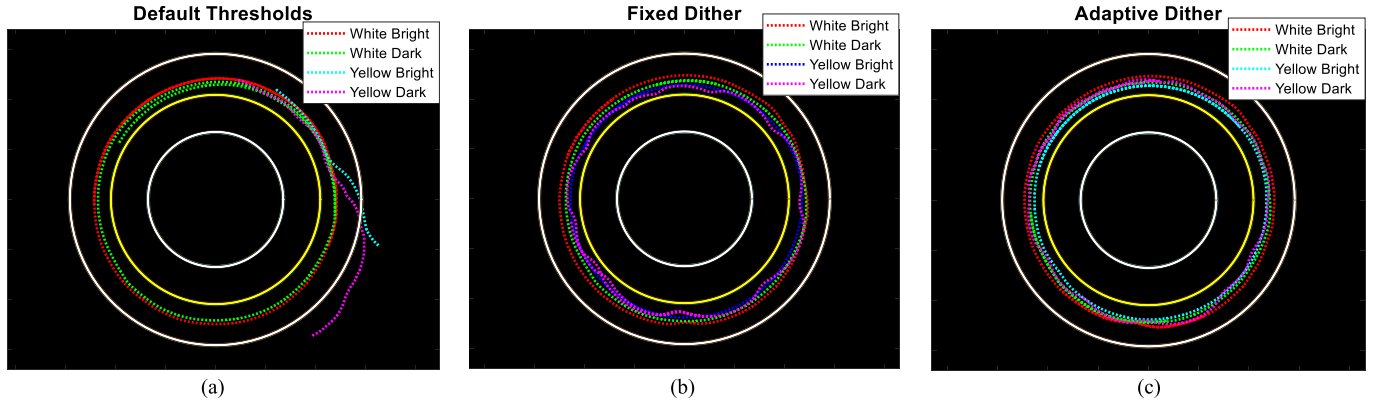


Fig. 9. Lane following results from the indoor GPS. Solid lines are lane marks, and dashed lines are vehicle paths. It shows that the scaled car went off the track with improper thresholds when following the yellow lane but successfully followed the white lane even without implementing the ES adjustments.

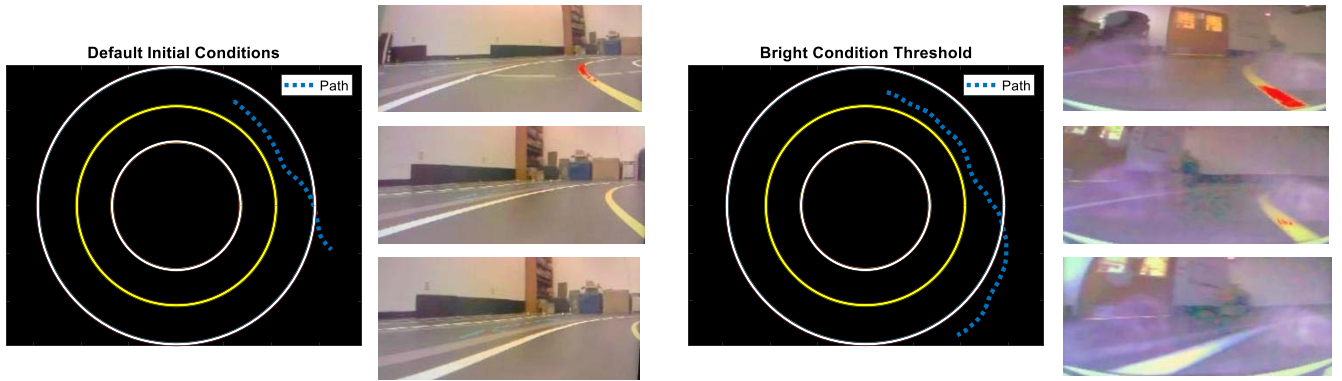


Fig. 10. Real-time video captures for the yellow lane following. The region marked by red pixels is the identified lane mark. Captured images demonstrate that the vehicle cannot perform good lane identification, and it gradually goes off the track.

TABLE III
ADJUSTED THRESHOLDS FOR THE FIELD TEST

		White Lane Detection						Yellow Lane Detection					
		H_{min}	S_{min}	V_{min}	H_{max}	S_{max}	V_{max}	H_{min}	S_{min}	V_{min}	H_{max}	S_{max}	V_{max}
Fixed Dither	Bright	0	0	171.5	254.8	53.18	255	0	55.24	167.9	97.24	255	255
	Dark	0	0	171.1	255	75.27	255	0	17.75	160.1	127.4	255	255
Adaptive Dither	Bright	0	0	168.4	255	55.24	254.8	0	67.84	171.6	106.3	255	255
	Dark	0	0	174.3	255	63.8	255	0	40.78	166.5	110.9	255	255

Thresholds in the S and V channels converge to $[0, 60]$, and $[170, 255]$, respectively. Low saturation (S) and high value (V) are consistent with the white color representation in the HSV space. For the yellow lane detection, the H channel converges to $[0, 105]$, which is in the vicinity of the yellow color. The S channel converges to $[60, 255]$, which is approximately the complement of the white color. It is expected since a higher saturation leads to a purer color. The range in the V channel is higher under the bright condition since it is a measure of the color brightness.

Lane-following results obtained from the indoor GPS are shown in Fig. 9. Solid lines represent lane marks, and dashed lines represent the scaled car's trajectory. Fig. 9 (a) shows that the scaled car follows the white lane under the bright condition with the default thresholds, which is consistent with the low costs shown in Fig. 8 (a). However, this will no

longer be true if we change the environment to what we used for the simulation. As shown in Fig. 5 (a), the initial guess is inaccurate with more background noises. By inspecting Table III, we find that the thresholds for white lane detection under bright and dark conditions are close to each other. This observation is consistent with our conjecture from the simulation that the white headlights make white lane marks look similar in an image under both lighting conditions. Thus, it is reasonable to see that the scaled car can successfully follow the white lane under different lighting conditions with the same thresholds, as shown in Fig. 9 (a).

Unlike the white lane detection, Fig. 9 (a) shows that the scaled car quickly loses tracking of the yellow lane marks. It can be explained by the poor performance of the default thresholds shown in Fig. 8 (c). Table III also shows that the thresholds under the bright condition are inappropriate for

the dark condition. Furthermore, Fig. 10 visualizes the failure of the yellow lane detection with “incorrect” thresholds. The identified pixels are masked with red color. Under both lighting conditions, we can see that initially only a small part of the lane marks is identified. After a short time, very few pixels can be recognized.

With the adjusted thresholds from the ES methods, Fig. 9 (b) and (c) indicate that the scaled car succeeds in staying within the lane. It is hard to tell the difference in the lane-following abilities between the two ES schemes, and this is expected since Fig. 8 shows that the cost function converges to a similar point from both methods. The superiority of the adaptive dither ES is its ability to drive the control variable to a smaller region around the local extremum which is helpful when choosing the adjusted system input as there is less oscillation.

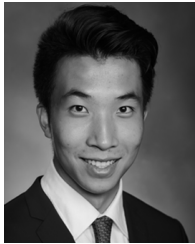
VI. CONCLUSION

This paper designs two ES based methods for autonomous vehicle lane detection under changing lighting conditions and demonstrates their effectiveness. Simulation studies and a field test indicate that lane identification accuracy is higher than 97.8%. The proposed algorithm is learning-free and computationally efficient compared to the machine learning based methods. The scaled car successfully followed an indoor circular map autonomously with the help of ES based lane detection methods. However, it failed to do so with manually defined thresholds or thresholds defined from the mismatched lighting conditions. Additional filters or more carefully chosen ROI are needed if there are objects on the road with a similar color to the lane marks, which will be studied more thoroughly in the future. Due to the scope of this study, limitations of the indoor space, and environmental requirements of the indoor GPS, only three lighting conditions were tested. Future studies will focus on testing for more lighting conditions on a full-size vehicle platform.

REFERENCES

- [1] R. Utriainen, M. Pollanen, and H. Liimatainen, “The safety potential of lane keeping assistance and possible actions to improve the potential,” *IEEE Trans. Intell. Vehicles*, vol. 5, no. 4, pp. 556–564, Dec. 2020, doi: [10.1109/ITV.2020.2991962](#).
- [2] 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 Recognit.*, vol. 73, pp. 216–234, Jan. 2018, doi: [10.1016/j.patcog.2017.08.014](#).
- [3] A. Borkar, M. Hayes, and M. T. Smith, “Robust lane detection and tracking with ransac and Kalman filter,” in *Proc. 16th IEEE Int. Conf. Image Process. (ICIP)*, Nov. 2009, pp. 3261–3264, doi: [10.1109/ICIP.2009.5413980](#).
- [4] P.-C. Wu, C.-Y. Chang, and C. H. Lin, “Lane-mark extraction for automobiles under complex conditions,” *Pattern Recognit.*, vol. 47, no. 8, pp. 2756–2767, Aug. 2014, doi: [10.1016/j.patcog.2014.02.004](#).
- [5] R. Mehrotra, K. R. Namuduri, and N. Ranganathan, “Gabor filter-based edge detection,” *Pattern Recognit.*, vol. 25, no. 12, pp. 1479–1494, Dec. 1992, doi: [10.1016/0031-3203\(92\)90121-X](#).
- [6] R. O. Duda and R. E. Hart, “Use of the Hough transformation to detect lines and curves in pictures,” *Commun. ACM*, vol. 15, no. 1, pp. 11–15, Jan. 1972, doi: [10.1145/361237.361242](#).
- [7] K. Zhao, M. Meuter, C. Nunn, D. Müller, S. Müller-Schneiders, and J. Pauli, “A novel multi-lane detection and tracking system,” in *Proc. IEEE Intell. Vehicles Symp.*, Jun. 2012, pp. 1084–1089, doi: [10.1109/IVS.2012.6232168](#).
- [8] C. Lee and J.-H. Moon, “Robust lane detection and tracking for real-time applications,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 12, pp. 4043–4048, Dec. 2018, doi: [10.1109/ITITS.2018.2791572](#).
- [9] R. K. Satzoda, S. Sathyanarayana, and T. Srikanthan, “Hierarchical additive Hough transform for lane detection,” *IEEE Embedded Syst. Lett.*, vol. 2, no. 2, pp. 23–26, Jun. 2010, doi: [10.1109/LES.2010.2051412](#).
- [10] D. Neven, B. D. Brabandere, S. Georgoulis, M. Proesmans, and L. V. Gool, “Towards end-to-end lane detection: An instance segmentation approach,” in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2018, pp. 286–291, doi: [10.1109/IVS.2018.8500547](#).
- [11] Z. Li, L. Yao, X. Chang, K. Zhan, J. Sun, and H. Zhang, “Zero-shot event detection via event-adaptive concept relevance mining,” *Pattern Recognit.*, vol. 88, pp. 595–603, Apr. 2019, doi: [10.1016/j.patcog.2018.12.010](#).
- [12] S. Lee *et al.*, “VPGNet: Vanishing point guided network for lane and road marking detection and recognition,” in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Oct. 2017, pp. 1947–1955.
- [13] Y. Hou, Z. Ma, C. Liu, and C. C. Loy, “Learning lightweight lane detection CNNs by self attention distillation,” in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2019, pp. 1013–1021.
- [14] Z. Qin, H. Wang, and X. Li, “Ultra fast structure-aware deep lane detection,” in *Proc. Eur. Conf. Comput. Vis.*, Aug. 2020, pp. 276–291, doi: [10.1007/978-3-030-58586-0_17](#).
- [15] Y.-C. Hsu, Z. Xu, Z. Kira, and J. Huang, “Learning to cluster for proposal-free instance segmentation,” in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2018, pp. 1–8, doi: [10.1109/IJCNN.2018.8489379](#).
- [16] Z. Li, F. Nie, X. Chang, L. Nie, H. Zhang, and Y. Yang, “Rank-constrained spectral clustering with flexible embedding,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6073–6082, Dec. 2018, doi: [10.1109/TNNLS.2018.2817538](#).
- [17] Z. Li, F. Nie, X. Chang, Y. Yang, C. Zhang, and N. Sebe, “Dynamic affinity graph construction for spectral clustering using multiple features,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6323–6332, Dec. 2018, doi: [10.1109/TNNLS.2018.2829867](#).
- [18] J. Yan, S.-H. Baek, and S.-Y. Park, “Robust road line color recognition based on 2-dimensional S-color space,” in *Proc. IEEE Int. Conf. Multisensor Fusion Integr. for Intell. Syst. (MFI)*, Nov. 2017, pp. 342–347, doi: [10.1109/MFI.2017.8170452](#).
- [19] Y. Zhou, Z. Wang, and J. Wang, “Real-time adaptive threshold adjustment for lane detection application under different lighting conditions using model-free control,” *IFAC-PapersOnLine*, vol. 54, no. 20, pp. 147–152, 2021, doi: [10.1016/j.ifacol.2021.11.167](#).
- [20] S. Jung, J. Youn, and S. Sull, “Efficient lane detection based on spatiotemporal images,” *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 289–295, Jan. 2016, doi: [10.1109/ITITS.2015.2464253](#).
- [21] M. Li, Y. Li, and M. Jiang, “Lane detection based on connection of various feature extraction methods,” *Adv. Multimedia*, vol. 2018, pp. 1–13, Aug. 2018, doi: [10.1155/2018/8320207](#).
- [22] N. J. Killingsworth and M. Krstic, “PID tuning using extremum seeking: Online, model-free performance optimization,” *IEEE Control Syst.*, vol. 26, no. 1, pp. 70–79, Feb. 2006, doi: [10.1109/MCS.2006.1580155](#).
- [23] M. Bagheri, M. Krstić, and P. Naseradinmousavi, “Multivariable extremum seeking for joint-space trajectory optimization of a high-degrees-of-freedom robot,” *J. Dyn. Syst., Meas., Control*, vol. 140, no. 11, Nov. 2018, doi: [10.1115/1.4040752](#).
- [24] J. Creaby, Y. Li, and J. E. Seem, “Maximizing wind turbine energy capture using multivariable extremum seeking control,” *Wind Eng.*, vol. 33, no. 4, pp. 361–387, Jun. 2009, doi: [10.1260/030952409789685753](#).
- [25] O. Zou and R. Y. Zhong, “Automatic logistics in a smart factory using RFID-enabled AGVs,” in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2018, pp. 822–826, doi: [10.1109/AIM.2018.8452349](#).
- [26] E. H. C. Harik, F. Guerin, F. Guinand, J.-F. Brethe, and H. Pelvillain, “Towards an autonomous warehouse inventory scheme,” in *Proc. IEEE Symp. Ser. Comput. Intell. (SSCI)*, Dec. 2016, pp. 1–8, doi: [10.1109/SSCI.2016.7850056](#).
- [27] D. Paddeu, I. Shergold, and G. Parkhurst, “The social perspective on policy towards local shared autonomous vehicle services (LSAVS),” *Transp. Policy*, vol. 98, pp. 116–126, Nov. 2020, doi: [10.1016/j.tranpol.2020.05.013](#).
- [28] M. Kamal, M. Atif, H. Mujahid, T. Shanableh, A. R. Al-Ali, and A. Al Nabulsi, “IoT based smart bus stops,” in *Proc. 3rd Int. Conf. Smart Grid Smart Cities (ICSGSC)*, Jun. 2019, pp. 6–9, doi: [10.1109/ICSGSC.2019.00-27](#).

- [29] B. Hunnekens, A. Di Dino, N. Van De Wouw, N. Van Dijk, and H. Nijmeijer, "Extremum-seeking control for the adaptive design of variable gain controllers," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 3, pp. 1041–1051, May 2015, doi: [10.1109/TCST.2014.2360913](https://doi.org/10.1109/TCST.2014.2360913).
- [30] W. H. Moase, C. Manzie, and M. J. Brear, "Newton-like extremum-seeking for the control of thermoacoustic instability," *IEEE Trans. Autom. Control*, vol. 55, no. 9, pp. 2094–2105, Sep. 2010, doi: [10.1109/TAC.2010.2042981](https://doi.org/10.1109/TAC.2010.2042981).
- [31] A. Ghaffari, M. Krstić, and D. Nešić, "Multivariable Newton-based extremum seeking," *Automatica*, vol. 48, no. 8, pp. 1759–1767, Aug. 2012, doi: [10.1016/j.automatica.2012.05.059](https://doi.org/10.1016/j.automatica.2012.05.059).
- [32] M. Krstić and H.-H. Wang, "Stability of extremum seeking feedback for general nonlinear dynamic systems," *Automatica*, vol. 36, no. 4, pp. 595–601, Apr. 2000, doi: [10.1016/S0005-1098\(99\)00183-1](https://doi.org/10.1016/S0005-1098(99)00183-1).
- [33] D. C. Andrade *et al.*, "A novel strategy for road lane detection and tracking based on a vehicle's forward monocular camera," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 4, pp. 1497–1507, Apr. 2019, doi: [10.1109/TITS.2018.2856361](https://doi.org/10.1109/TITS.2018.2856361).
- [34] S. Chen, L. Huang, H. Chen, and J. Bai, "Multi-lane detection and tracking using temporal-spatial model and particle filtering," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 3, pp. 2227–2245, Mar. 2022, doi: [10.1109/TITS.2020.3035614](https://doi.org/10.1109/TITS.2020.3035614).



Yujing Zhou received the B.S. degree (*magna cum laude*) in mechanical engineering from Bucknell University, Lewisburg, PA, USA, in 2020. He is currently pursuing the M.S. degree with the Walker Department of Mechanical Engineering, The University of Texas at Austin, Austin, TX, USA. His research interests include robotics, vehicle dynamics and control, and data-driven control.



Austin, TX, USA. His research interests include vehicle dynamics and control, combustion engine control, and cyber-physical systems.

Zejiang Wang (Graduate Student Member, IEEE) received the B.E. degree (Hons.) in mechanical engineering and automation from Southeast University, Nanjing, China, in 2014, the Dipl.-Ing. degree from ENSTA ParisTech, Palaiseau, France, in 2017, and the M.S. degree in design, modeling, and architecture of complex industrial systems from the École Polytechnique, University of Paris-Saclay, Palaiseau, France, in 2017. He is currently pursuing the Ph.D. degree with the Walker Department of Mechanical Engineering, The University of Texas at Austin,



Steen Norris Endowed Professor at the Walker Department of Mechanical Engineering, The University of Texas at Austin. He is also an IEEE Vehicular Technology Society Distinguished Lecturer, an SAE Fellow, and an ASME Fellow.

Junmin Wang (Senior Member, IEEE) received the B.E. degree in automotive engineering and the first M.S. degree in power machinery and engineering from the Tsinghua University, Beijing, China, in 1997 and 2000, respectively, the second M.S. and third M.S. degrees in electrical engineering and mechanical engineering from the University of Minnesota, Minneapolis, MN, USA, in 2003, and the Ph.D. degree in mechanical engineering from The University of Texas at Austin, Austin, TX, USA, in 2007. He is currently the Lee Norris and Linda