2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Constraint Decision Optimization Model for Safe Autonomous Vehicle Operation

H. Bryan Riley and Mehmet Celenk
School of Electrical Engineering and Computer Science
Athens, Ohio, 45701 USA

Abstract— **Numerous researchers and organizations continue to refine technologies that will soon allow vehicles to autonomously drive safely from Point A to Point B. Accuracy determining the position of the self-driving vehicle relative to lane markers and road boundaries specifically during inclement weather conditions continues is of primary importance. This paper presents an investigation and associated results where road land boundary markers are detected in conjunction with the ability decipher the horizon when the front view of the vehicle's path is degraded. Degradation of driving scenes can be attributed to such weather conditions as heavy rain, fog, snow, or dust storms. The detection of lane markers and road boundaries is especially important for roads that exhibit severe curves, aggressive uphill slopes and downhill valleys, respectively. Additionally, we present a model to predict deviations from reference distances associated with roads with such design constraints along with some promising experimental results.**

Keywords – Intelligent vehicles, Autonomous vehicle, IR sensors, Automated Driving Systems (ADS)

I. MOTIVATION AND BACKGROUND

The safe operation of self-driving vehicles is predicted based on the exact location of the vehicle relative to other traveling vehicles, lane markers, road boundaries and most importantly in-path objects. False positive assessment of any of these conditions or operating scenarios may cause injury to the driver, vehicle occupants or result in collision and property damage [1, 2, 3, 7,19]. More than several traffic dense cities have emerged as leaders in autonomous vehicle testing and innovation plus involvement of a combination of hi-tech corporations, cutting edge university research and enthusiastic political support [10]. Major contributors to accurate determination of vehicle position are dependent upon the road and the environmental conditions. Figure 1 seeks to present these conditions in search of fitting autonomous driving model of with Figure 2 as inputs to a constrained decision optimization model that makes informed decision for the vehicle control processor as it issues commands to the steering, braking and powertrain / throttle actuators. A review of the literature indicates that sensor studies relative to autonomous vehicle control and *lane-keeping* during adverse driving condition continues to be important. The literature indicates considerable research efforts are devoted to this area. L. Guang et .al [16] present the development of an autonomous vehicle following control scheme using a laser scanning radar sensor (LIDAR) for automated highway applications. Although LIDAR sensor measures the target vehicle's relative position with respect to its preceding vehicle these measurements degrade in inclement weather conditions [16]. A sophisticated implementation of obstacle and lane detection for a self-guided autonomous vehicle demonstrates success by designing computational and vison techniques for single instruction, multiple data (SIMD) computers. The cost here is an expensive architecture which is typical parallel computers in Flynn's taxonomy [17]. The remaining of the paper is organized as follows. Section II presents development of real-world driving model and constraint decision optimization. Section III describes algorithms designed for lane marker and edge detection. Section IV is devoted to embedded systems relative to detailed requirements and design for this application. Section V provides a description of the system implementation and testing of this investigation**.** Section VI presents an analysis of the test data and performance results. Section VII contains the conclusion recommendations for future work.

Figure 1. Various inclement weather and driving conditions resulting from wet pavement, icy pavement, sleet, snow or heavy fog.

II. DEVELOPMENT OF REAL-WORLD DRIVING MODEL AND CONSTRAINT DECISION OPTIMIZATION

In developing real world driving model, we consider the true values of the road conditions measured by multiple sensors and calibrated by a multisensory data acquisition system as

978-1-5386-4451-5/18/$31.00 ©2018 IEEE

autonomous vehicle travels toward its destination. Liu et al. study an optimization based on two-level control framework for autonomous vehicles in a receding horizontal fashion [18]. We consider the measured road conditions and associated parameters at the *time (t)* include the distance d(t) between the self-driving vehicle and the vehicle in same lane $d_1(t)$, the right adjacent lane $d_2(t)$ and the vehicle in the left adjacent lane $d_3(t)$. Effect of hazardous road conditions is included into the above-mentioned distances as distractive components (e.g., noise) as follows:

$$D_1(t) = d_1(t) + n_1(t) \quad 1(a)$$

$$D_2(t) = d_2(t) + n_2(t) \quad 1(b)$$

$$D_3(t) = d_3(t) + n_3(t) \quad 1(c)$$

where $D_1(t)$, $D_2(t)$ and $D_3(t)$ are the observed (measured) distances; $d_1(t)$, $d_2(t)$, and $d_3(t)$ are the actual distance (i.e., ground truth) values; and $n_1(t)$, $n_2(t)$, and $n_3(t)$ are the disturbances. The task here is to estimate the model distance values $\hat{d}_1(t), \hat{d}_2(t)$, and $\hat{d}_3(t)$ from the observed values $D_1(t)$, $D_2(t)$, and $D_3(t)$ by minimizing the disturbances $n_1(t)$, $n_2(t)$, and $n_3(t)$ whose appearances are presented in Figure 2 and Figure 3 respectively.

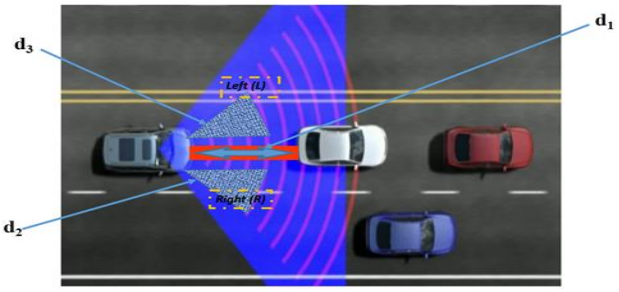

Figure 2. Real-world driving scenario where the host vehicle is autonomously following a target vehicle.

By rewriting Eqns 1(a), 1(b), and 1(c), we have

$$n_1(t) = D_1(t) - d_1(t) \quad 2(a)$$

$$n_2(t) = D_2(t) - d_2(t) \quad 2(b)$$

$$n_3(t) = D_3(t) - d_3(t) \quad 2(c)$$

Using the Least Mean Square (LMS) error criterion, we write the error Eqns $E_1(t)$, $E_2(t)$, and $E_3(t)$ as a function of time *(t)* as in Eqns 3(a), 3(b), and 3(c), respectively:

$$Error_1 = E[|n_1(t)|^2] \quad 3(a)$$

$$Error_2 = E[|n_2(t)|^2] \quad 3(b)$$

$$Error_3 = E[|n_3(t)|^2] \quad 3(c)$$

where *E* is the expected value operator.

This signal estimation problem is solved using the orthogonality principal which dictates that $n_1(t)$, $n_2(t)$, and $n_3(t)$ are uncorrelated with any random variables $D_1(t)$, $D_2(t)$, and $D_3(t)$. As a result, we state that

$$E[n_1(t)\, D_1(t)] = 0 \quad for\ all\ t \quad 4(a)$$

$$E[n_2(t)\, D_2(t)] = 0 \quad for\ all\ t \quad 4(b)$$

$$E[n_3(t)\, D_3(t)] = 0 \quad for\ all\ t \quad 4(c)$$

Then the solution is:

$$R_{d_1D_1}(t) = h_1(t) * R_{D_1}(t) \quad 5(a)$$

$$R_{d_2D_2}(t) = h_2(t) * R_{D_2}(t) \quad 5(b)$$

$$R_{d_3D_3}(t) = h_3(t) * R_{D_3}(t) \quad 5(c)$$

where $R_{d_iD_i}(t)$, i=1,2,3, are the cross and self-correlation functions, and $h_1(t),\ h_2(t), and\ h_3(t)$ are the impulse response functions of the unknown predictors. Taking the Fourier transform of both sides Eqns 5(a), 5(b), and 5(c) and carryout some algebraic manipulations [6] we obtain the frequency responses of the prediction filters as given in Eqns 6(a), 6(b), and 6(c).

$$H_1(\omega) = \frac{P_{d_1}(\omega)}{P_{d_1}(\omega)+P_{n_1}(\omega)} \quad 6(a)$$

$$H_2(\omega) = \frac{P_{d_2}(\omega)}{P_{d_2}(\omega)+P_{n_2}(\omega)} \quad 6(b)$$

$$H_3(\omega) = \frac{P_{d_3}(\omega)}{P_{d_3}(\omega)+P_{n_3}(\omega)} \quad 6(c)$$

where $P(\omega)'s$ are the power spectrum of the respective stochastic variable.

The system processing block diagram depicting the relationship between *D(t)* and $\hat{d}(t)$ is shown in Figure 3.

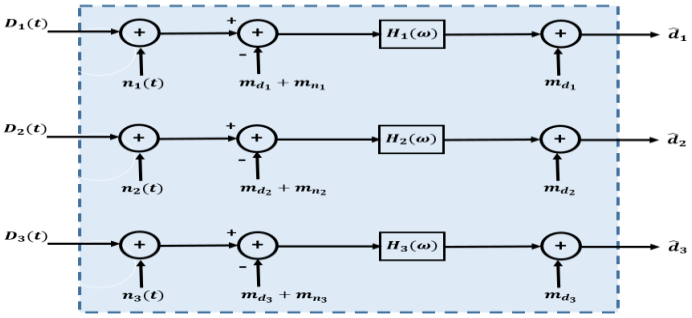

Figure 3. System processing block diagram where m_{d_i} (i=1,2,3) and m_{n_j} (j=1,2,3) are the means of the respective distances (D_1,D_2,D_3) and disturbances (n_1,n_2,n_3)

III. LANE MARKER AND LANE EDGE DETECTION

Refer to Figure 1 and consider it as a real-world rural road as shown in Figure 4. Figure 4 depicts a two lane road with lane markers indicating the LEFT **(L)** centerline road markers and RIGHT **(R)** shoulder natural boundary markings that are being tracked by dynamic local windows.

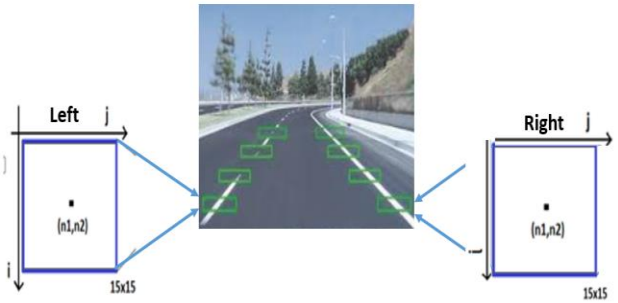

Figure 4. Graphical representation of a two lane road with dynamic local windows experimental determined to be optimal at 15x15 unit less dimensions.

Parajuli et al. [13] proposed this new method for lane detection which uses the frequency domain features of the lane markings. This work carried out extensive experimentation on a significantly large number of images where the dimensions were varied for a fixed number of trials. Results clearly indicated the windows are designed to be 15 x 15 pixels and robust to invariant translation, rotation, and scaling. Larger dimensioned windows tended to produce results of variation. Additional research and computer simulation results yielded a 95% successful rate using the images from [12]. Real world autonomous vehicle driving scenarios must account for environmental conditions (i.e., wind, rain, snow, ice, etc.), vehicle suspension variations, tire rubber/tread quality, road surface conditions, and the driver's mental state. However, in this study we consider the lateral and distance parameters. This is primarily due to the test bed utilized to study left and right lane markings when affected by such erroneous highly irregular disturbances as shadows and highlights. While the hazardous weather and road conditions are assumed to be included in the model's $n_1(t)$, $n_2(t)$, and $n_3(t)$ degradation parameters of Figure 3, respectively. The sensor data of points **L** and **R** are provided to the Gradient Spectrum Method (GSM) during the initialization step. For comparisons the Principal Component Analysis (PCA) method is also implemented. Next the local dynamic window of size 15x15 is placed at the points **L** and **R**. The local windows' major axis orientation is computed at points **L** and **R** on the horizontal scan line. The lane points on the next scan line are calculated in the direction of the window's major axis orientation from the preceding scan lines. Details of the algorithms and computations are provided in [13]. Figure 5 shows the result of GSM, and PCA method along with the horizontal scan line on the top left and middle column as indicated. The top right column is the result obtained after the imperative information is selected from the two methods (PCA and GSM). The computed and predicted points for left and right lane markings on each horizontal scan line (n) are plotted in Figure 5 for both the PCA and GSM method. The error between the prediction and computed points is computed and as shown in Figure 5. The predicted points, correspond to the minimum error in a mean square sense between the PCA and GSM method. Algorithms implement in this work are executed on the same data as referenced by Chang et at. [14]. Results are selected for each horizontal scan line per the computations and shown in final results.

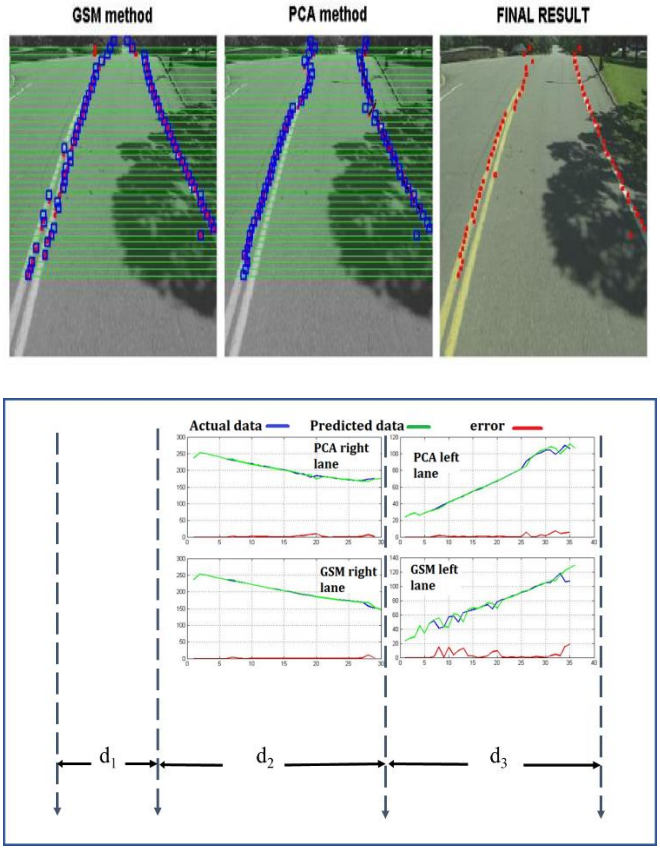

Figure 5: Results of the GSM method, PCA method and information selection and the respective error curves on lane segments. The red, blue points on the image results from GSM and PCA method are detected and predicted points respectively. The final image containing red points are the final result from the relevant information based upon the error curves. The error curves are plotted as a function of column no Vs horizontal scan line number, n.

IV. DESCRIPTION OF EMBEDDED SYSTEM DESIGN

This section describes the design and implementation of an embedded system onto 1/10 scaled remote controlled (RC) robotic vehicles. The system design for this embedded system began by conducting tradeoff study between the Arduino and Raspberry Pi single board computers. Base on ease of programing, a large suite software algorithm readily available

in the open source domain the decision was made to select the Raspberry Pi Model B^+. This is a full-featured, compact computer, operating at 700 MHz and executing the Raspbian Operating System (OS). This single board computer interfaces consist of:

- Four USB 2.0 Ports,
- Single SD card slot
- 24 Pin Header with GPIO, and
- An Ethernet connector,
- A CSI connector for a camera, and
- An HDMI output
- Power: supplied by micro-USB Connector

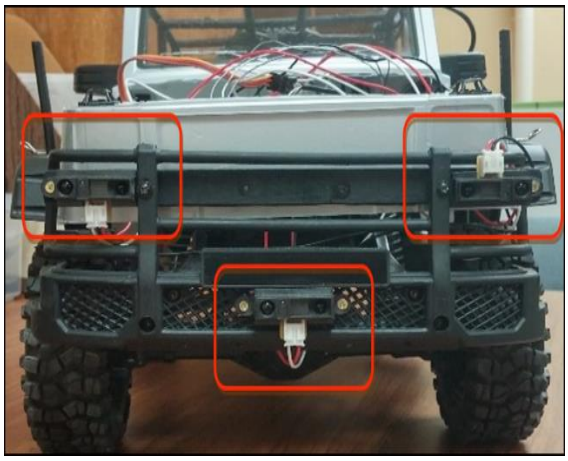

Figure 6. IR sensors mounted on 1/10 scaled RC vehicle.

The transducer, is a Sharp GP2YOD21YK IR sensor selected which operates at the IR wavelength of λ=780 nm [11]. It is an active data acquisition transducer consisting of a transmitter and a receiver unit both actively engaged transmitting and receiving operations continuous time. Figure 6 shows the mounting and configuration for 3 IR sensors to provide the forward looking operational FOV. The sensors are cost effective, easy to install, calibrate and operate, low voltage, less energy consumption, nonlinear operation characteristics (i.e., distance inversely proportional to output voltage reading), effective in short distance measurements, and not very effective in long distance measurements [8].

In this research, we model the operational characteristics of IR sensors in accordance with the physical principal as the measured distance d_i is inversely proportional (α) to the IR sensor voltage reading v_i; namely,

$$d_i \; \alpha \; \frac{1}{v_i} \quad , \mathrm{t} = 1,2,3. \tag{7}$$

Further, inverse proportionality is expressed as

$$d_i = k \frac{1}{v_i^a} \quad , \mathrm{t} = 1,2,3. \tag{8}$$

where k is a scaling constant variable and a is the degree of proportionality. Both of these constant variable are to be determined empirically in system implementation and experimentation. Since the Sharp GP2YOA21YK IR sensor was selected, the generalized mathematical relationship between distance and output voltage is given by

$$d = \left(\frac{1}{c * A_{DC} + b}\right) - K \tag{9}$$

where d is distance cm, K is a corrective constant, A_{DC} is the digitized value of the sensor output voltage, b denotes a variable constant value to be determined from trend line Eqn, and c denotes another variable constant to be determined from the trend line equation. Notice that our proposed Eqn (2) has only one constant variable (k) to be determined whereas the governing sensor Eqn (3) has three variables (b, c, and K) that must be determined empirically from the measurement data. To this end, it is concluded that our model Eqn (2) accounts for a multiple sensor configuration and is easier to use and implement than the sensor governing Eqn (3), respectively as the RC vehicle convoy from a random starting location to another location.

V. Testing and Analysis

This embedded system is defined by time-division multiplexing principles since the sensor data streams are formatted into a single signal by separating the signal into short duration segments. The sensor data is collected and stored in memory for processing. Each of the Sharp IR distance measuring sensors are independently selected after several tests to benchmark performance and verify the electrical interface with the Raspberry Pi Model B^+ single board computer. V_{cc} is the source voltage and this device can tolerate a range of 4.55v to 5.5v. Figure 7 show the electrical interconnection block diagram of the Sharp IR sensor where the digital output V_o is directed to the digital I/O channel of a Raspberry Pi Model B^+ single board computer. [12], [13].

The system is designed such that the path of travel contains minimal deviations in radius of curvature, the acceleration, and speed are modeled by control variables for the following experiments. The software for IR sensing system as depicted in Figure 7 is written in Python version 2.7 software. The algorithm is designed per Proportional Integral Derivative controller (PID controller) for this closed loop system. For this application the PI algorithm is selected as the most instinctive, superior, and simplistic. [15] The servo motor speed of the robotic RC vehicle is the process variable. The expected tracking performance is be smooth. The principle of this algorithm is based on manipulating the error (e) which is defined as the difference between the Process Variable (PV) and the Set Point (SP). Figure 8 depicts the PI controller block diagram. The defined headway or fixed distance for this vehicle controller is defined by the set point "SP" and computed distances, which are independent of the measurement sensor "PVs". The MATLAB R2017A

function "PIDSYS" is implemented and returns the continuous-time PI controller in parallel form.

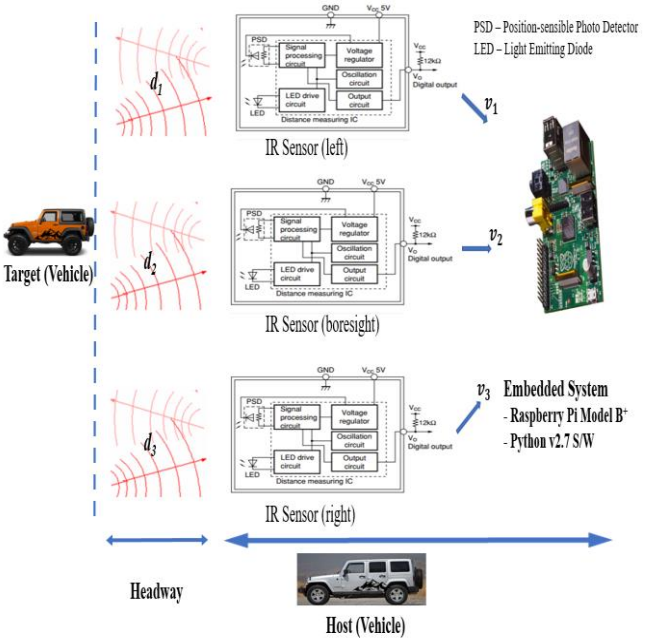

Figure 7. Embedded system for autonomous convoy.[14].

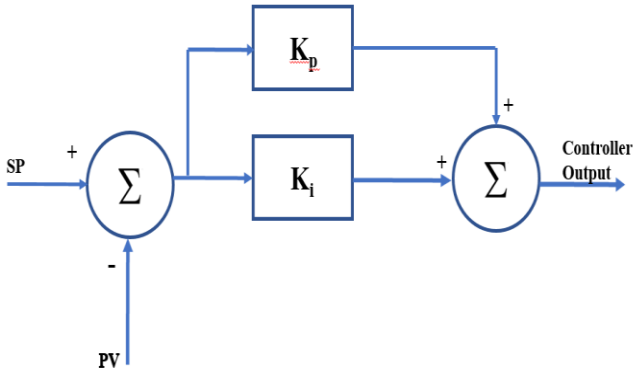

Figure 8. PI controller block diagram.

Proportional (P) corrects instances of error and the Integral (I) corrects the accumulation of error. As e(t) increases or decreases, the amount added to the Controller Output (CO) increases or decreases immediately and proportionately. Mathematically, the CO is governed by Eqn 4 and is given by

$$Controller_{output} = \beta \cdot \int e(\tau)\, d\tau \qquad (10)$$

where β is the proportional gain and e(τ) are defined to compute the controller output. Based on sensor voltages defined in Figure 7 and computing distances, independent of sensor specification, tracking by the host vehicle occurs per Eqn 5.

$$d_1 > (d_2 + d_3) \Rightarrow \text{host tracks target to left} \qquad (11a)$$

$$d_1 \simeq d_2 \simeq d_3 \Rightarrow \text{host tracks target straight} \qquad (11b)$$

$$d_3 > (d_1 + d_2) \Rightarrow \text{host tracks target to right} \qquad (11c)$$

Upon successful implementation and debugging of the PI software the control loop is tuned or adjusted by starting with low proportional and no integral, continue to double the proportional until it begins to oscillate, then reduce by ½. Next incorporate a small integral and double it until it starts to oscillate, then reduce it by ½. Continue to determine a reasonable headway distance (headway) and smooth following behavior as the hosts track the target. MATLAB "PIDSYS" returns the proportional and integral controller terms as K_p = 1.14 and K_i = 0.454 respectively. The corresponding step response for this PI controller is computed and the error or deviation of actual measured PV from the SP is shown in Figure 9.

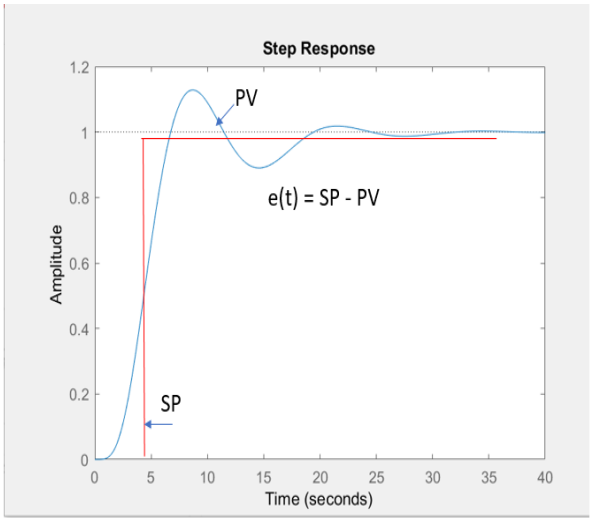

Figure 9. Step response for PI controller.

VI. Analysis of Test Results

Performing trial runs with the embedded system on the host vehicle showed not jerking effect and smoothly tracked the target.

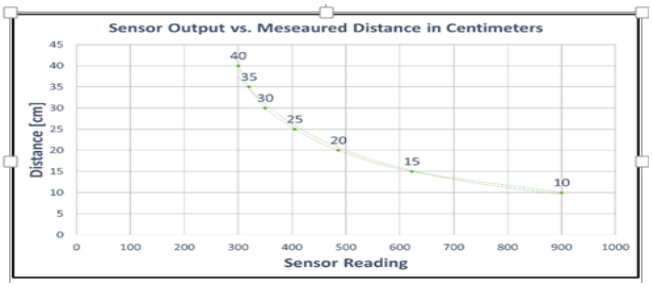

Figure 10. Sensor output vs. distance (cm).

During several test trials, the change in sensor reading from 5cm to 10cm was much greater than the change in sensor reading from 30 cm to 35 cm. To account for the non-linear characteristics of the IR sensors the embedded software was modified to approximate the exponential curve that was computed to be the performance. The host RC at different distances ranging from 10 cm to 40 cm while measuring the sensor output. Uniform testing was conducted per outline procedures and the data was plotted in Microsoft Excel. The plot results are shown in Figure 10. A Microsoft Excel data function is used to calculate an exponential trend line and is given by Eqn (6):

$$d_i = 43{,}000\ v_i^{-1.236} \quad \iota = 1.2., 3 \tag{12}$$

where d_i is the measured distance in cm and v_i indicates the IR sensor readings of the digital output. Review of the curve in Figure 3 for an actual case indicates for $v_a = 900 \Rightarrow d_a$=10cm. Alternatively the estimated distance is

$$\widehat{d_a} = 43000(900)^{-1.236} = 9.594\ \chi\mu \tag{13}$$

The average error Ɛ in %, is given Eqn (8) and in this case

$$\text{Ɛ} = \frac{(\hat{d}_a - d_i)}{\hat{d}_a}(100) \Rightarrow 4.054\% \tag{14}$$

A resulting headway (i.e., following distance) is maintained to less than 5% error and the steady state response is measured to be on the order of 8 seconds. A comparison to other research results reported in the literature confirms similar result where the trade of is use of two sensors (i.e., narrower FOV) whereas this work utilized a configuration of three sensors ((i.e., broader FOV). Suggested next steps for this research shall include replicating real-world environmental conditions (i.e., fog) for testing and analysis of the "lead vehicle" and "sensor vehicle". Suggested future work is to investigate an approach consists resolving distances based on the speed of light by measuring the time-of-flight (ToF) between sensors and targeted image using a ToF Camera or LIDAR. Robust sensing technologies will be enables for major corporations such as Tesla, Mercedes-Benz, Ford, Google, General Motors, and Uber have boldly asserted they will have fully autonomous passenger cars and trucks on public roads by 2021.

where d_a is the actual distance in cm based on 3 Sharp model GP2YOD21YK forward looking IR sensors.

Recorded data is subjected to Eqn (1) via a MATLAB algorithm, converting the arbitrary sensor readings into real distance values. This led to the errors being generated in the PI loop to be linear with the distance and the control system's performance improved significantly. The MATLAB function "POLYFIT" is utilized to return the coefficients of a 6th order polynomial that is a best fit in the least-squares sense to the error data where it is scaled to have unit standard deviation.

VII. Conclusion

This paper proposes an embedded system design shown to be implemented and successfully tested to perform as a testbed for testing IR sensors for autonomous vehicle sensing applications. A system architecture that will allow of sensors of a varied range of specifications can be incorporated to ultimate demonstrate performance of maintaining a set headway and smoothly following a target vehicle. The hardware prototype including an imbedded processor was implemented and defined experiments were performed to demonstrate the performance of the proposed optimization model for safe autonomous vehicle operations.

ACKNOWLEDGMENTS

We wish to thank Camron Schuman and James Petersen for their work to instrument the RC vehicles with IR sensors and perform the data collection of $d_1(t)$ as depicted in Figure 3.

REFERENCES

[1] S. Tamer et al. "Autonomous Driving - The Impact of Vehicle Automation on Mobility Behavior", Institute for Mobility Research (ifmo), http://www.ifmo.de/publications.html?t=45 (2016).

[2] D.B. Maciuca and C.J. Gerdes, "*Automatic Braking Control for Intelligent Vehicle and Highway Systems (IVHS)"*, Proceedings of the International Symposium on Advanced Vehicle Control, 1994.

[3] W. Qui et al., "Autonomous vehicle longitudinal following control based on model predictive control", IEEE Conference on Control Applications (CCA), 103-108, 2017.

[4] Upton, E. and Halfacree, G., *Raspberry Pi User Guide*, Wiley Publication, 2012.

[5] Python Software Foundation, *Python 2.7.13 Website Address Documentation*, https://docs.python.org/2/

[6] B. Farhang-Boroujeny, Adaptive Filters Theory and Application 2nd edition Wiley, 2013.

[7] L. Zilong, C. Wenbo, and J. Yanxia, "Wide speed-range following control for road autonomous vehicles," Vol. 56, No. 6, EEE Transactions on Vehicular Technology, pp. 3660-3673, November 2007.

[8] MD10C 3.0 Motor Controller. http://cytron.com.my/p-md10c [Assessed: 27Oct2017].

[9] SHARP Part Number GP2Y0A21YK IR Distance Sensor. http://www.sharpsma.com/webfm_send/1489.

[10] "Cities vie to become hubs of self-driving technology", USA TODAY, 2017. [Online]. Available: https://www.usatoday.com/story/money/cars/2017/06/25/cities-vie-become-hubs-self-driving-technology/100963464/

[11] A. Crabb, "AdaSky Launches Thermal Far Infrared (FIR) Sensing Solution to Give Vehicles 24/7 Vision; Accelerates Fully-Autonomous Driving," Cision PR Newswire.

[12] Carnegie-Mellon-University, "CMU/VASC image database 1997–2003," http://vasc.ri.cmu.edu/idb/html/road/may30_90/

[13] A. Parajuli, M. Celenk, and H. Riley, "Robust lane detection in shadows and low illumination conditions using local gradient features," *Open Journal of Applied Sciences*, Vol. 3 No. 1B, 2013, pp. 68-74. doi: 10.4236/ojapps.2013.31B014.

[14] C. Chag et al., "An efficient method for lane-mark extraction in complex conditions," Ubiquitous Intelligent & Computing and 9th International Conference on Autonomous & Trusted Computing (UIC/ATC), IEEE, pp. 330-336. September 2014.

[15] R.E. Precup et al., "PI and PID controller tuning for an automotive application using backtracking search optimization algorithm", SAIC 2015, 10th Jubil IEEE International Symposium Application Computer Intelligence Informatics, Proceedings. 2015.

[16] L. Guang et al., A laser scanning radar based autonomous lateral vehicle following control scheme for automated highways, American Control Conference Proceedings. June 4-6, 2003.

[17] M. Bertozzi and A. Broggi, "Vision-based vehicle guidance" Computer, vol. 30, no.7, July 1997, pp.49-55.

[18] C. Liu, Wen-Hau Chen and John Andrews, "Optimisation based control framework for autonomous vehicle: algorithm and experiment," Proceedings of the IEEE, Aug. 4-7, 2010.

[20] ` M. Nolte, M. Rose, T. Stolte, and M. Maurer, "Model Predictive Control Based Trajectory Generation for Autonomous Vehicles – An Architectural Approach".