

# Line Estimation for a Line-Following Mobile Robot

Alpaslan Yufka<sup>1</sup> and Aydın Aybar<sup>1</sup>

<sup>1</sup>Department of Electrical and Electronics Engineering, Anadolu University, Eskişehir, Turkey  
ayufka@gmail.com, aaybar@anadolu.edu.tr

## Abstract

**In this study, the coordinate of the line position for a differential drive line-following mobile robot is estimated by using the Minimum Variance Unbiased Estimator. Using the sensor data as an observation data and coordinates of sensors, a linear model for Minimum Variance Unbiased Estimator is constructed. MATLAB simulations for computing Root-Mean-Square-Error and Monte Carlo Simulations and real world data are used to analyze the performance of the estimator. Using the estimated value, a feedback for PID controller is used to compute deviation in order to make the mobile robot follow the line.**

## 1. Introduction

A Line-Following Mobile Robot (LFR) is an autonomous non-holonomic mobile robot that is capable of tracking the line on the ground which is in contrast such as black line and white floor (or vice versa). LFR has not only Infrared (IR) Sensors to detect the line strip such that it steers itself with respect to data coming from sensors, but it may have also optical sensors, cameras, etc. to sense the line. These sensor outputs such as analog signals are used in the microcontroller in order to execute the line-following algorithm. As a result of this algorithm, necessary Pulse-Width-Modulation (PWM) signals are generated in order to drive DC motors such that the control algorithm so as to keep LFR on the line.

The main problem of the line-following is that how LFR smoothly and accurately follows the line. To perform this, the more LFR has knowledge about the line position according to the center of it, the better it tracks. The difference between the line position and LFR provides it to compute the deviation as an error input for the controller. There are many techniques to calculate this. In [1], using phototransistor array, a fuzzy logic technique with six inputs is used for finding the line position according to the wheeled mobile robot as one output. In [2], seven analog infrared sensors are used to locate the coordinate of the center of line. Using three of them which give higher outputs than others, a quadratic equation is used and as a result of this equation the coordinate of the parabola's peak gives the coordinate of the central line. In addition, a calibration technique is used for these analog sensors. In [3], eight reflective optical sensors are used to find the correct position of the line. A quadratic interpolation technique which is as same as in [2] is introduced and the coordinates of consecutive optical sensors with higher output are used as inputs of this equation. In [4], a single IR sensor which is composed of an IR Light Emitting Diode (LED) and a phototransistor is used to detect line where two color shades such as grey and black on the white floor such that the intensity value against position on line is obtained. In

[5], 32 phototransistors are used to detect the white guide lines on the dark blue ground such that each sensor outputs on the white line is sampled by 1000 times and the average of them is computed as an sample value. Using sample values, a threshold value for both the white line and the dark ground is calculated.

In this work, a sensor array included 8 IR sensors is used for detecting the line so that sensor outputs as analog inputs for the microcontroller are measured. Each sensor has a unique adjacent position-coordinate. These local coordinates of sensors construct the observation matrix which is known and analog measurements of IR sensor array gives the data observation vector whose size is as same as the number of sensors in use. The environmental noise is supposed to be White Gaussian Noise (WGN) whose mean is zero. Using the observation matrix, the data observation and noise vector of WGN, a linear model is formed as the Minimum Variance Unbiased (MVU) Estimator that allow LFR to readily determine the coordinate of the line's position so that it computes the deviation according to the coordinate of the line's center by using the estimated line's position. This deviation becomes an input for the Proportional-Integral-Derivative (PID) Controller as an error term. Then, MATLAB [9] simulations for computing Root-Mean-Square-Error (RMSE) and Monte Carlo Simulations and real world data are carried out in order to analyze the performance of the estimator.

In Section II, the mathematical model for constructing the linear model and defining MVU estimator, and ID control method is explained. In Section III, applications such as simulations and the real world data are realized by using MATLAB and LFR's sensor array. In Section IV, the study is concluded briefly.

## 2. Methodology

### 2.1. Linear Model

In this project, the IR Sensor Array of LFR includes 8 QRD1114 Reflective Object Sensors which has a phototransistor output [6, 7]. The output of these sensor array is digitized by 8-bit Analog-to-Digital Converter (ADC) in the microcontroller called as Peripheral Interface Controller (PIC) 16F887 [8]. Measurements of sensors are denoted as  $m_1, m_2, m_3, m_4, m_5, m_6, m_7,$  and  $m_8$ . The position of leftmost sensor which is  $p_1$  is taken into account 0, and then the position of rest is consecutively increased by 9.5 millimeter such as  $p_2$  is 9.5 mm,  $p_3$  is 19 mm,  $p_4$  is 28.5 mm,  $p_5$  is 38 mm,  $p_6$  is 47.5 mm,  $p_7$  is 57 mm, and  $p_8$  is 66.5 mm. The width of the line strip is  $19 \pm 1$  mm, and the center of LFR's sensor array shall be on the middle of the line strip such that the reference coordinate of central line is 33.25 mm which is used as a set point for the controller. The coordinates of sensor's locations and their possible outputs

where LFR is on the middle of the line strip is illustrated in Fig.1.

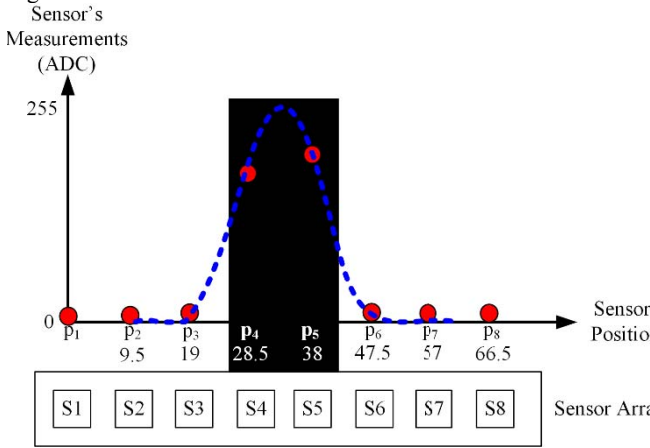


Fig. 1. Locations of Sensors and Their Outputs

Noise vector denoted as  $N=[n_1 \ n_2 \ n_3 \ n_4 \ n_5 \ n_6 \ n_7 \ n_8]^T$  where  $[.]^T$  indicates the transpose of  $[.]$  is considered as WGN whose mean is zero. The estimator is construct by a quadratic equation whose coefficients are unknown and desired to estimate such that these parameters is represented by  $\theta=[A \ B \ C]^T$ . Using these definitions, the line estimation problem is represented by a linear model given in (1) and (2).

$$m_i = p_i^2 A + p_i B + C + n_i, \text{ where } i = 1, 2, \dots, 8 \quad (1)$$

$$\begin{bmatrix} m_1 \\ \vdots \\ m_8 \end{bmatrix} = \begin{bmatrix} p_1^2 & p_1 & 1 \\ \vdots & \vdots & \vdots \\ p_8^2 & p_8 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} + \begin{bmatrix} n_1 \\ \vdots \\ n_8 \end{bmatrix} \quad (2)$$

Due to computational time and width of the line's strip, 4 sensors (2 of 4 have higher outputs than others, and other 2 is adjacent to them) are considered for the line estimation problem. These are taken into account as the data observation vector represented as  $X=[x_1 \ x_2 \ x_3 \ x_4]^T$ , and the observation matrix is denoted as  $H$  whose size four by four. The noise vector for the related sensors is represented as  $W=[w_1 \ w_2 \ w_3 \ w_4]^T$ . The reduced linear model is given in (3).

$$\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} h_1^2 & h_1 & 1 \\ h_2^2 & h_2 & 1 \\ h_3^2 & h_3 & 1 \\ h_4^2 & h_4 & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} + \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \quad (3)$$

$$X = H\theta + W \quad (4)$$

Using (3) and (4), the Minimum Variance Unbiased Estimator (MVU) is given in (5).

$$\hat{\theta} = (H^T H)^{-1} H^T X \quad (5)$$

Using (5), the estimated unknown parameters are found. These parameters  $A$  and  $B$  are used for estimating the coordinate of line position such that the maximal point of the parabola (taking first derivative of the quadratic equation and equals it to zero) gives this coordinate where Equation (6) is used for computing this.

$$\text{The Estimated Coordinate of Line Position} = -\frac{B}{2A} \quad (6)$$

## 2.2. Proportional-Integral-Derivative Controller

PID Controller is used to generate necessary control inputs for LFR's DC motors in order to make LFR follow the line strip accurately. This controller consists of three constants which are proportional (P), integral (I) and derivative (D). The traditional representation of PID controller is given in (7), and the block diagram of PID controller for LFR is shown if Fig. 2.

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{d}{dt} e(t) \quad (7)$$

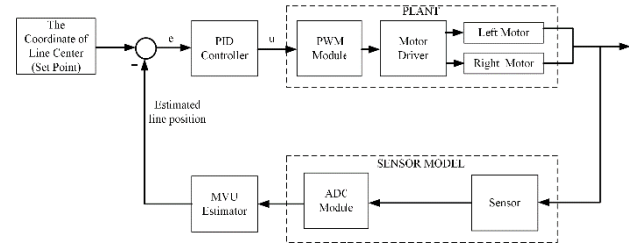


Fig. 2. Control Block Diagram for LFR

Equation (7) is a continuous-time function. To imply this controller to LFR, it should be rewritten in discrete-time as given in (8).  $k$  denotes the discrete-time variable.

$$\begin{aligned} P &= K_p e(k) \\ D &= K_d (e(k) - e(k-1)) \\ I &= K_i (e(k) + \sum_{l=0}^{k-1} e(l)) \\ u(k) &= P + I + D, \text{ for } k \geq 0 \end{aligned} \quad (8)$$

In (8),  $P$  term is computed by using proportional constant and the error which is the deviation of the estimated coordinate of the line position according to the coordinate of the sensor array's center which presents the set point. The error is computed as given in (9).  $D$  term is approximately calculated using the previous and the current error.  $I$  term is approximately is considered as the summation of errors to compute the overall system's error. Due to the small deviation in a certain time (in robot races, duration of one lap is sufficiently short) and the small integral constant,  $I$  term can be neglected.

$$e(k) = \begin{matrix} \text{the coordinate of} \\ \text{center (33.25 mm)} \end{matrix} - \begin{matrix} \text{the estimated} \\ \text{coordinate of} \\ \text{the line position} \end{matrix} \quad (9)$$

The output of the PID controller becomes an input of the PWM module in the microcontroller. This module generates required PWM signals to the right and the left motors for the differential drive. Then, the PWM module transmits signals to the motor driver of LFR in order to manage motors.

## 3. Application

Applications are carried out in two ways such as a simulation and real-world experiments.

### 3.1. Simulation

The simulation is performed by using MATLAB. First of all, 1000 independent zero-mean white Gaussian random variable for  $\sigma=6$  dB is generated for each sensors. The mean and the variance are found numerically. The Probability Density Function (PDF) is drawn by using a histogram as shown in Fig. 3.

Data of Sensor outputs with WGN is generated for each sensors as shown in Fig. 4. These data is only generated for the line where LFR on the middle of the line strip. Using these sensor outputs, the data observation vector is approximately found as  $m=[7.8 \ 8.162 \ 15.3 \ 7.7 \ 8]^T$  by calculating mean value of the sensor data. Outputs of fourth and fifth sensors are obviously sufficiently larger than others such that these two and their adjacent sensors are considered in order to estimate line position. Thus, the data observation vector is reduced to  $X=[8.162 \ 15.3 \ 7]^T$ . Related sensors are 3<sup>rd</sup>, 4<sup>th</sup>, 5<sup>th</sup> and 6<sup>th</sup> sensors whose positions are consecutively  $p_3=19$ ,  $p_4=28.5$ ,  $p_5=38$ , and  $p_6=47.5$ . These build the observation matrix  $H$  by using (1)-(3) as in (10).

$$H = \begin{bmatrix} 361 & 19 & 1 \\ 812.25 & 28.5 & 1 \\ 1444 & 38 & 1 \\ 2256.3 & 47.5 & 1 \end{bmatrix} \quad (10)$$

Using (5), the estimated unknown parameters are found as in (11).

$$\hat{\theta} = \begin{bmatrix} -0.8317 \\ 55.2 \\ -739.52 \end{bmatrix} \quad (11)$$

Using (6), estimated line position is found as 33.185 mm where the real position of the line is 33.25 mm. The error between the estimated value and the real value is less than 0.1 mm which is reasonable.

In addition to simulation, the Root-Mean-Square-Error (RMSE) performance of the MVU estimator using the averages of  $M = 250$  Monte Carlo simulations for the noise levels  $\sigma \in [-10 : 1 : 10]$  dB and  $\sigma \in [-10 : 1 : 50]$  dB are obtained and plotted as shown in Fig. 5 and Fig. 6 respectively.

RMSE values in Fig. 5 and Fig. 6 are small in different noise levels. This means that MVU estimator using for the line detection is robust to changes in noise level. Sensor array is affected together, but the linear model still works. If larger noise levels make sensors saturated, the line can be detected incorrectly.

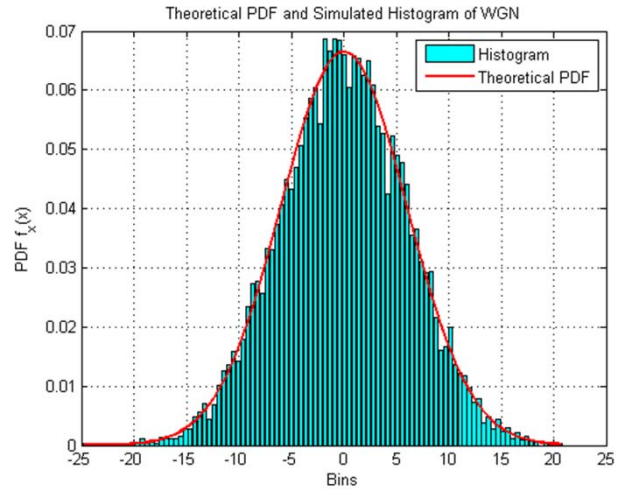


Fig. 3. Simulated Histogram of WGN and Theoretical PDF

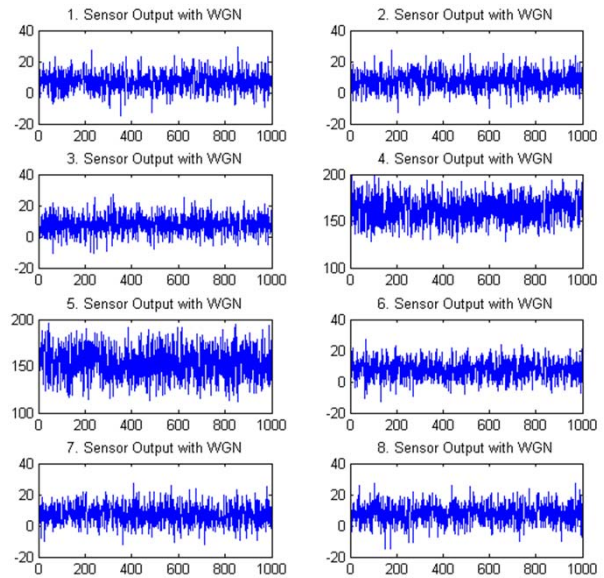


Fig. 4. Sensor Outputs with WGN

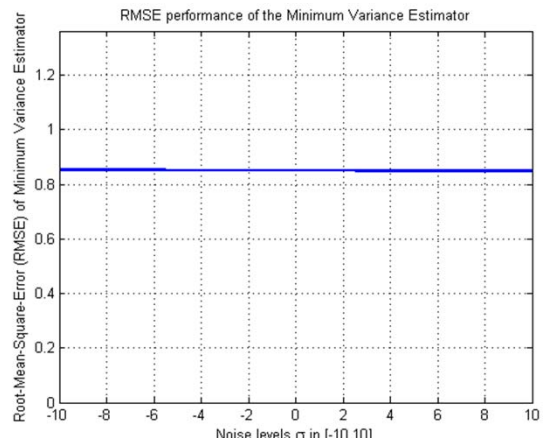


Fig. 5. The RMSE performance of the MVU estimator where  $\sigma \in [-10 : 1 : 10]$  dB

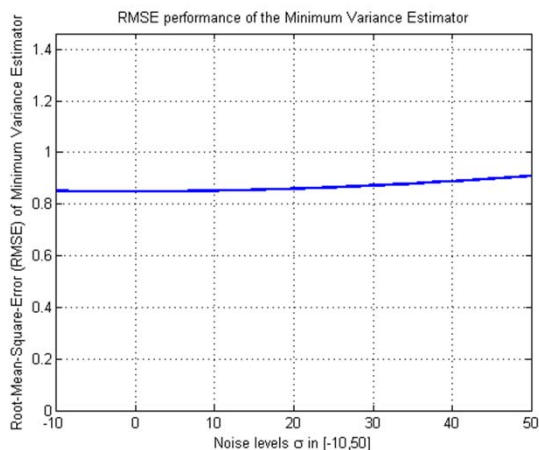


Fig. 6. The RMSE performance of the MVU estimator where  $\sigma \in [-10 : 1 : 50]$  dB

### 3.2. Real World Data

The real world application is performed by using a sensor array which includes 8 QRD1114 Reflective Object Sensors. These sensors are analog such that 8-bit ADC module (values between 0 and 255) in the microcontroller PIC 16F887 is used for digitizing the analog signal and 2x16 characters LCD display is used to read ADC values of the sensor array. 10 samples of each sensor’s measurements are taken for three cases: LFR **i**) on the middle of the line strip, **ii**) 20 mm left, **iii**) 20 mm right. Then, the mean value of these samples is computed for each cases as given in Table I. It gives the data observation vector such that these are used for analyzing in the MATLAB.

Table 1. Mean Value of the Measurements for Each Sensor

Case	S1	S2	S3	S4	S5	S6	S7	S8
i)	6	7	7	163	162	7	9	9
ii)	10	194	79	7	8	8	10	10
iii)	9	8	8	7	8	136	200	11

The data observation vector from Table I is reduced to  $X_1=[7 \ 163 \ 162 \ 7]^T$  where related sensors are 3<sup>rd</sup>, 4<sup>th</sup>, 5<sup>th</sup> and 6<sup>th</sup>,  $X_2=[8 \ 136 \ 200 \ 11]^T$  where related sensors are 5<sup>rd</sup>, 6<sup>th</sup>, 7<sup>th</sup> and 8<sup>th</sup>, and  $X_3=[10 \ 194 \ 79 \ 7]^T$  where related sensors are 1<sup>st</sup>, 2<sup>nd</sup>, 3<sup>rd</sup> and 4<sup>th</sup>. The matrix H is found by using (1)-(3), and the estimated unknown parameters are obtained by using (5). Estimated line position is found as:

- 1) For case i) It is found as 33.2449 where the real position of the line is 33.25 mm. The error is less than 0.01.
- 2) For case ii) It is found as 52.6781 where the real position of the line is 53.25 mm. The error is less than 0.6 mm.
- 3) For case iii) It is found as 13.3286 where the real position of the line is 13.25 mm. The error is less than 0.1 mm.

In this section, a mean of sensor measurements is taken for MATLAB simulations in order to make analyze readily. In the real world application, LFR does not take a mean of sensor measurements. It uses sensor measurements directly in the linear

model in order to estimate the coordinate of the line’s position in real time.

### 4. Conclusion

The coordinate of the line’s position is estimated by using the MVU Estimator. Using the estimated line’s position as a feedback for the PID controller, the deviation according to the coordinate of the sensor array’s center is computed. This deviation is used for generating necessary PWM signals in order to drive differentially LFR’s DC motors.

The simulations and the real-world data show the MVU estimator gives admissible results, and RMSE performance of the estimator is sufficiently small even if noise level is large. On the other hand, the difference between the estimated line’s position and the real line’s position is reasonably small such that the output of MVU estimator is approximately as same as the real line’s position. It shows that the estimator is unbiased.

Future work may not only include different type of estimator and their comparison among each other, but it may also include predict the next coordinate of the line’s position. It may also include a proposal of a novel estimator. To support the study, the real world application of LFR may be carried out.

### 5. References

- [1] K.H. Ng, C.F. Yeong, E.L.M. Su, T.Y. Lim, Y. Subramaniam, and R.S. Teng, “Adaptive Phototransistor Sensor for Line Finding”, International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012), vol. 41, pp. 237-243, 2012.
- [2] F.H. Jen and B.T. Mai, “Building an autonomous line tracing car with PID algorithm”, IEEE Intelligent Control and Automation (WCICA), 2012 10th World Congress on, pp. 4478 - 4483, July 2012.
- [3] M. Engin and D. Engin, “Path planning of line follower robot”, IEEE Education and Research Conference (EDERC), 2012 5th European DSP, pp. 1-5, September 2012.
- [4] N.M. Arshad, M.F. Misnan, and N.A. Razak, “Single Infra-Red Sensor Technique for Line-Tracking Autonomous Mobile Vehicle”, Signal Processing and its Applications (CSPA), 2011 IEEE 7th International Colloquium on, pp. 159-162, March 2011.
- [5] Y. Li, X. Wu, D. Shin, W. Wang, J. Bai1, Q. He1, F. Luo, and W. Zheng, “An Improved Line Following Optimization Algorithm for Mobile Robot”, IEEE Computing and Convergence Technology (ICCCT), 2012 7th International Conference on, pp. 84-87, December 2012.
- [6] FAIRCHILD Semiconductor, Datasheet of QRD1113/1114 Reflective Object Sensor, Available: <http://www.solarbotics.net/library/datasheets/QRD1114.pdf>, Last Accessed : 11/08/2015.
- [7] Pololu Robotics&Electronics, Datasheet of QTR-8RC Reflectance Sensor Array, Available: <http://www.pololu.com/docs/pdf/0J12/QTR-8x.pdf>, Last Accessed : 11/08/2015.
- [8] Microchip Technology Inc., Datasheet of PIC16F882/883/884/886/887 Microcontroller, Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/41291D.pdf>, Last Accessed : 11/08/2015.
- [9] The MathWorks, Inc., MATLAB Software, Available: <http://www.mathworks.com/products/matlab/>, Last Accessed : 11/08/2015.