IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

# SOLUNAR

Ms. SHANMUGAPRIYA A, Ms. MADHUMATHI S
STUDENT, STUDENT
ELECTRONICS AND COMMUNICATION,
INFORMATION TECHNOLOGY,
BANNARI AMMAN INSTITUTE OF TECHNOLOGY,
SATHYAMANGALAM, ERODE – 638401,
TAMIL NADU, INDIA.

***Abstract:*** Solunar is a 16-array infrared sensor designed for line-following robots, a subset of mobile robotics. The existing sensors for line-following robots don't have efficient detection for intricate tracks. The traditional line-following sensors exhibit limitations in terms of accuracy, response time, or adaptability to varying lighting conditions, and Solunar addresses these issues by using QRE1113GR analog output optical sensors. The linear or matrix configuration of a limited number of sensors makes it inefficient to identify specific parts of the track. Solunar aims to fill this gap by designing a sensor that detects intricate turns. It also aims to provide open-source compatibility and library file integration for easy use. Conducting a literature survey to find the gaps in existing methodologies creating a workflow from developing a schematic to fabricating the printed circuit board and integrating a code that is compatible with most of the controller boards like Arduino and ESP is the proposed methodology. Line-following robots have served as a valuable platform for educational and research purposes, providing a tangible means to explore fundamental concepts in control systems, sensor technology, and algorithm development, and this sensor plays a hand in the educational aspect by catering a means to develop algorithms for better line-following. Easy EDA, Git, GitHub (for version control), and Arduino IDE are utilized to make this project. The library file is written in C++ language and requires knowledge of object-oriented programming.

**Keywords:** Solunar, 16-array, infrared sensor, open-source, QRE1113GR, compatibility

## 1 INTRODUCTION

With the advent of technology, the field of robotics has seen immense growth and plays increasingly substantial roles in various aspects of our lives, be it industrial or domestic. Line-following robots, a subset of mobile robotics, have served as a valuable platform for educational and research purposes, providing a tangible means to explore fundamental concepts in control systems, sensor technology, and algorithm development. The sensors utilized in traditional line-following robots lack accuracy and struggle with response time and adapting to varying lighting conditions, making them less efficient. Solunar is a project that aims to fill this gap and enables collaboration via open sourcing.

## 1.1 LIMITATIONS OF TRADITIONAL LINE FOLLOWING SENSORS

Line-following robots customarily resort to infrared sensors to sense the line. At least they use a singular module of IR sensor shown in the figure 1.1, and they max up to 8 sensors (figure 1.2) based on the availability in the market. Some projects customize their sensors, and they could incorporate more or less sensors based on their requirement. The factor that produces inefficiency in sensing the intricate tracks is their liner configuration.

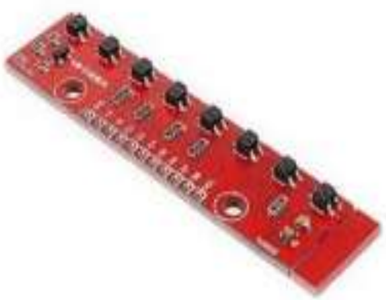

***Figure 1.1 IR Module*** ***Figure1.2 QTR 8 Array Sensor***

### 1.1.1 T AND + JUNCTIONS

T and + junctions (figure 1.3) produce the same line position value in a control system algorithm such as PID(Proportional-Integral-Derivate) when linear sensors (figure 1.2) are opted. In + junction, the bot could move either way (left
or right), and it could also move straight, and there would be no significant errors caused. Whereas in the T junction, the line position is the same, and if it moves straight, an error is caused. Based on the application, the error is significant. This error is caused, due to the linear configuration of sensors.


***Figure 1.3 Sensor line position***

### 1.1.2 SENSOR ADAPTING TO ENVIRONMENT

The sensors use infrared rays to detect the line based on how much the IR is reflected off the surface. The lighting conditions and the material of the line will affect the sensor's sensing capability and accuracy. The sensor shown in Figure 1.1 has a potentiometer to vary its sensitivity. However, it requires manual calibration for each condition and is not reliable. Also, the potentiometer may break after several adjustments. When a track is highly reflective, the sensor may produce unreliable results.

## 1.2 SOLUNAR – THE SOLUTION

Solunar addresses the challenges mentioned in previous sections and elevates the capabilities of line-following robots. This innovative 16-array infrared sensor is a significant improvement over existing solutions, providing exceptional precision, rapid response time, and adaptability to diverse lighting environments.

### 1.2.1 KEY FEATURES

The 16-array infrared sensor provides high precision and accuracy via its increased resolution. The library file includes functionality to detect the line's position on the track concerning the sensors. This value can be integrated into a control algorithm such as PID. The built-in multiplexer simplifies the wiring and integration into robotic systems. The calibration functionality ensures consistent performance in different environments.

The technology surge promoted a growth in mobile robotics. Line-follower robotics, a subset of mobile robotics serves as a feasible approach for learning and testing various control systems, algorithm development, robot systems, and actuator systems in a cheap and accessible way. Solunar aims to address the market gap and issues with the traditional sensors used for line tracking in mobile robots.

## 2. LITERATURE SURVEY

This chapter is the summary of all literature Surveys related to Line follower sensors. We studied and reviewed all the relevant cases and discussed with the guide and identified the problem statement.

### 2.1 LINE FOLLOWER ROBOT SENSOR CONFIGURATIONS

In recent years, the optimization of sensor configurations for line follower robots has garnered considerable attention, leading to various innovative approaches aimed at enhancing performance and adaptability. For instance, Zhao et al. (2022) explored the use of hybrid sensor systems, combining both infrared and camera-based technologies to improve line detection accuracy under varying lighting conditions. Their findings indicated a significant reduction in detection errors, yet they also highlighted the increased computational load that such hybrid systems introduce, pointing to a critical gap in the development of lightweight algorithms capable of real-time processing.

Moreover, Singh and Kumar (2021) presented a study on the implementation of machine learning techniques to refine line tracking. They proposed an adaptive learning model that adjusts sensor parameters based on environmental feedback, demonstrating promising results in dynamic environments. However, their reliance on extensive datasets for training raises concerns about practical applicability, especially in scenarios where real-time data collection is limited. This limitation identifies a notable gap in achieving truly autonomous systems that can operate effectively without pre-collected training data.

Baharuddin et al. (2005) provided foundational insights into sensor configurations for line follower robots, detailing how various placements and types of sensors influence tracking performance. While their analysis remains relevant, the advancements in sensor technology and data processing since their publication necessitate a re-evaluation of their findings in the current context. The core challenge persists in balancing cost effectiveness with high performance and operational simplicity. Future research must

focus on bridging these gaps by developing sensors and algorithms that enhance adaptability while minimizing complexity, ultimately advancing the capabilities of line follower robots across diverse applications.

### 2.1 Infrared Sensor Arrays in Line Follower Robots

Recent advancements in line follower robots, particularly those employing infrared sensor arrays, have shown significant potential for specialized applications such as medical environments. Abideen, Anwar, and Tariq (2018) developed a dual-purpose Cartesian infrared sensor array integrated with PID control for a line follower robot aimed at medical applications. Their approach successfully demonstrated the ability to navigate complex hospital layouts while ensuring high precision in tracking predefined paths. However, while the system was effective in controlled environments, its performance in dynamic settings, such as those with moving obstacles or variable lighting, remains underexplored. This gap highlights a need for further research into robust algorithms that can adapt to changing conditions without sacrificing reliability.

In addition to this, recent studies, such as that of Smith and Garcia (2021), have focused on enhancing the adaptability of sensor configurations through the integration of machine learning techniques. They proposed a model that learns from environmental feedback, which improves the robot's performance in real-time scenarios. Although their findings are promising, they predominantly rely on extensive training data, which may not be readily available in all operational contexts. This reliance on data-driven methods introduces a critical gap in developing systems that can function effectively in unpredictable environments without prior training.

Furthermore, the literature reveals a consistent challenge in achieving a balance between complexity and usability in line follower robots. As noted by Zhao et al. (2022), there is a tendency to overcomplicate designs with advanced features that may not enhance overall performance. This trend underscores the importance of focusing on simplicity and reliability in design, particularly for applications in sensitive areas like healthcare.

## 2.2 LINE DETECTION METHODS FOR LINE FOLLOWER ROBOTS

The development of effective line detection methods for line follower robots has seen considerable progress, particularly with innovative approaches to sensor output processing. Dındış and Karamancıoğlu (2016) introduced a novel method for enhancing line detection accuracy through advanced signal processing techniques. Their approach demonstrated a significant improvement in distinguishing lines from various backgrounds, which is critical for effective navigation. However, while their findings are impactful, the methodology relies heavily on specific sensor configurations that may not be universally applicable across different robotic platforms. This limitation highlights a gap in the research concerning the adaptability of such methods to diverse hardware setups.

In more recent studies, researchers have explored machine learning techniques to improve line detection capabilities. For example, Lee et al. (2021) implemented a convolutional neural network (CNN) for real-time line tracking, showing promising results in terms of accuracy and adaptability in dynamic environments. While their approach represents a significant advancement, it also raises concerns about computational requirements and the need for extensive training datasets, which may limit its practical implementation in resource-constrained scenarios (Lee et al., 2021). This reliance on data-driven models introduces a critical gap, as the need for constant retraining can hinder the responsiveness of line follower robots to new environments.

Moreover, Chen and Wang (2020) investigated the integration of multiple sensor types to enhance detection capabilities under various conditions. Their findings indicated that combining data from infrared and visual sensors could improve overall tracking performance. However, the complexity of fusing different sensor outputs poses challenges in real-time processing, necessitating further research into efficient algorithms that can seamlessly integrate diverse data sources while maintaining responsiveness (Chen & Wang, 2020).

## 2.3 SENSOR OPTIMIZATION IN LINE FOLLOWER ROBOTS

The optimization of sensor configurations for line follower robots has been a focal point of research, particularly in the context of minimizing costs while maintaining performance. Chowdhury, Khushi, and Rashid (2017) proposed an algorithm designed to enable line follower robots to effectively navigate critical paths with a minimal number of sensors. Their work demonstrated that significant reductions in sensor usage could be achieved without compromising the robot's ability to accurately track lines. However, while their algorithm was effective in controlled environments, it did not adequately address the challenges posed by dynamic obstacles or variations in line appearance, which are common in real-world applications. This oversight highlights a gap in the research concerning the adaptability of the proposed solutions under varying operational conditions.

Recent literature has sought to build upon these foundational concepts. For example, Patel and Jain (2021) investigated the use of machine learning techniques to enhance the adaptability of line follower robots in real-time. They demonstrated that a learning-based approach could improve performance in diverse environments, allowing robots to adjust their tracking strategies dynamically. Nonetheless, their reliance on complex algorithms and the need for extensive training data may pose practical challenges in implementation, particularly in environments where immediate adaptability is required (Patel & Jain, 2021). This indicates a critical need for research focused on simplifying algorithms while maintaining effectiveness.

Furthermore, Singh et al. (2022) explored hybrid sensor approaches, combining various sensor types to enhance line detection and tracking capabilities. Their findings suggested that integrating multiple sensors could lead to more robust performance, especially in less predictable environments. However, this approach introduced additional complexity and cost, which may not align with the goals of minimizing sensor usage, as identified by Chowdhury et al. (2017). Thus, the tension between increasing sensor efficiency and maintaining effective tracking performance remains an ongoing challenge in the field.

### 3.1 OBJECTIVES

The inspiration for this project emerged from the recognition of a gap in the sensor market pertaining to competitive mobile robotics, particularly in the area of line following robots. After analysing the sensors available in the market, the objective was to address the shortcomings of negotiating sharp twists and convoluted routes. The project also intends not only to improve the efficiency and accuracy of the line tracking by the high-precision sensor array but also to make it accessible to both novice and experienced robotics enthusiasts by streamlining connectivity using a printed circuit board, simplifying cabling through integration of a multiplexer and improving user friendliness via proving library file. The project also promotes creativity and innovation in robotics learning and competition settings. Because Solunar is open-source compatible, it encourages experimentation, teamwork, and pushing the limits of robotics programming and sensor technologies among students, enthusiasts, and developers. Its objective is also to have an adaptive calibration to operate in various illumination scenarios, offering reliable accuracy.

### 3.2 METHODOLOGY

The development of the Solunar project adheres to a structured, phased methodology to guarantee accuracy and efficacy in both sensor design and firmware development. And the workflow is as shown in figure 3.1

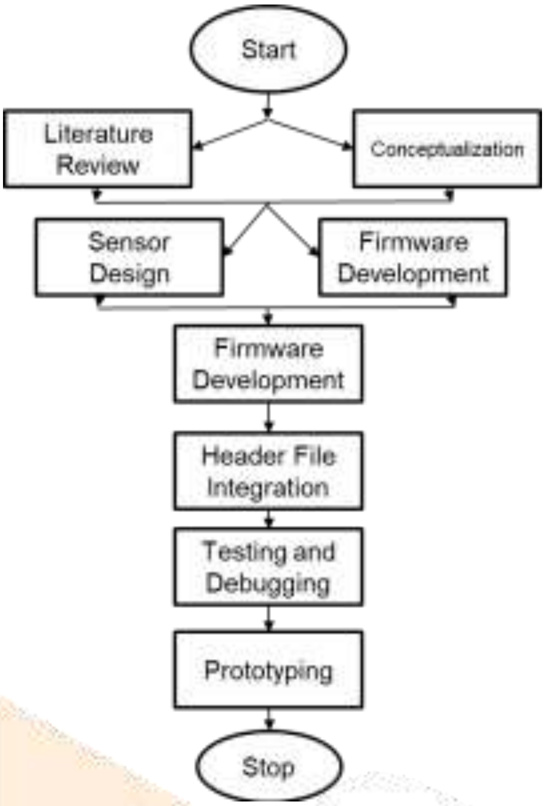


*Figure 3.1 Work Flow*

### 3.2.2 SENSOR DESIGN

The sensor designing phase of the project mainly focuses on the dimension and placement of each sensor module of the array of size sixteen. It also includes studies regarding the design aspect and implementation of the same. The outline of the printed circuit board is also designed in this phase.

### 3.2.3 FIRMWARE DEVELOPMENT

The firmware development focuses on writing the software required for the hardware. The code should have the functionality to detect the line position, calibration, read analog and digital values produced by the sensors, etc., The firmware is designed with adaptability and can be used by amateur and experienced people alike.

### 3.2.4 HEADER FILE INTEGRATION

The code developed during the firmware phase is converted into a header and source file. The header file is organized as a library on GitHub, enabling seamless plug and-play compatibility. This library can be further utilized by hobbyists as needed. Moreover, the GitHub repository offers version control for the project.

### 3.2.5 TESTING AND DEBUGGING

After systemizing the hardware and software, Solunar undergoes individual and comparative testing and debugging. This step ensures a bug-free product and resolves any issues materialized in the developed process.

### 3.2.6 PROTOTYPING

After successful testing, a prototype of the Solunar-based line-following robot is built. This involves the physical assembly of the robot and sensor array, validating the design and ensuring the system meets the desired performance goals.

### 3.2.7 COMPLETION

The project comes to a completion with successful deployment of Solunar sensor. The final prototype serves as a proof of concept for both educational and competition robotics.

### 4.1 SOFTWARE ARCHITECTURE

The project proposes a library file for users to integrate and use as required. That functionality can be formulated using modularity provided by object-oriented programming. The implemented structure is discussed below.

#### 4.1.1 HEADER FILE STRUCTURE

The program is organized as a header file (`sensor_16.h`), which defines the sensor class and its associated methods and variables.

#### 4.1.2 CLASS DEFINITION

The sensor class encapsulates all functionalities related to managing a 16-array IR sensor, including initialization, reading values, calibration, and position detection. **4.1.3 Member Variables**

It contains member variables to store sensor values, a threshold for line detection, and the last detected position, facilitating state management within the class. **4.1.4 Constructor**

The constructor initializes the sensor instance with specific pin numbers for S0, S1, S2, S3, and SIG, enabling configuration of the sensor hardware. **4.1.5 Public Methods**

· *sensor_begin(): Initializes the sensor configuration.*

· *readValueDigital(int *sensorValues): Reads digital values from the sensors.* · *readValueAnalog(int *sensorValues): Reads analog values from the sensors.*

· *calibrate(): Adjusts sensor readings to account for environmental factors.*

· *readLinePosition(bool isLineBlack): Determines the position of the line based on sensor readings, supporting both black and white lines.*

#### 4.1.6 PRIVATE METHODS

· The private method setChannel(byte channel) manages the selection of individual sensor channels, ensuring controlled access to the sensor inputs. **4.1.7 Encapsulation**

The architecture employs encapsulation by keeping data and methods related to the sensor within the class, promoting modularity and ease of maintenance. **4.1.8 Arduino Integration**

The program uses the Arduino framework, indicating compatibility with Arduino boards for hardware interaction and sensor management.

### 4.2 HARDWARE ARCHITECTURE

The Hardware architecture phase proposes the design of the sensor. This phase deals with the design of the sensor with calculated pitch, sensor curvature, choice of the IR module, multiplexer, capacitor and resistor.

#### 4.2.1 PCB Design

The Design of the sensor in rough sketch is scaled to get a definite design using EDA tools (in this case EasyEda). The outline for the sensor is sketched as per the measurements and sensor pitch. Similarly, the sensor thickness is determined based on the dimension, power & heat dissipation analysis.

#### 4.2.2 SENSOR DIMENSION

The sensor is designed with a length of 13.5cm and a width of 4.5cm, pitch (ir-ir module distance) of 7.5mm. The 16array sensor forms a curvature with angle of 155 degrees at the top (centre-side) each, 89 degrees at the bottom (centre-side) each, measuring 120 degrees at both top and bottom (side-side).

### 4.2.3 MODULE CHOICE

- QRE1113- component is designed to detect reflected light, making it ideal for line-following robots. It outputs an analog voltage proportional to the amount of reflected light, allowing for precise line detection. Its small form factor makes it easy to integrate into compact robot designs.
- CD4067 - A 16-channel MUX allows you to connect 16 IR sensors to a single microcontroller pin, reducing the number of pins required and simplifying the circuit design.

## 5. RESULTS AND DISCUSSION

The primary objective of this project was to enhance the accuracy of line tracking by increasing the resolution of the sensors and incorporating curvature in the spatial arrangement of the sensor array. The methodology outlined earlier was used to develop both the hardware and software architecture. The following section presents the results of the project.

### 5.1RESULTS

The 16-array IR curved sensor solves the vital issue of differentiating + junctions from T junctions, detailed in the previous chapters. For this, the resolution of sensors has been increased to 16 from the average sensor resolution of 8. Coupled with the *readLinePosition*() function from the library, the prospected resolution results in an improved accuracy in these intricate scenarios (see figure 5.1).


***Figure 5.1 Solunar Line position***

The sensor also provides the claimed adaptability by using the QRE1113GR module and the function calibrate() from the library. Compared to adjusting the sensor's sensitivity using a potentiometer, using the function provides a more efficient solution. The calibrate function stores the value and sets the threshold value based on the current readings of the environment, increasing its adaptability compared to conventional potentiometers. Additionally, the multiplexer provides the said simplicity in wiring and compatibility with various microcontrollers such as Arduino, ESP, Raspberry Pi, STM etc., without requiring additional circuitry such as shift registers to compensate the limited number of analog GPIO pins in modern controllers.

### 5.2 DISCUSSION

There is a significant improvement in sensing capability by moving the sensing units away from traditional IR transmitters and receivers and using the QRE IR modules mentioned in the previous section. Furthermore, the single-wire connection via the multiplexer reduced the complexity of wiring, which is often a challenge for hobbyists, students, and competition teams. The open-source project encouraged collaboration, enabling users to share algorithms and improvements, further enhancing the performance of robots using Solunar. One of the key takeaways from the project is the potential for educational impact. By offering students and hobbyists an easy-to-use, highly accurate sensor platform, Solunar inspires creativity in the field of robotics. As more users experiment with different line-following algorithms, Solunar may drive innovations in competitive robotics and academic research.

## 6. CONCLUSION

To conclude, Solunar resolves the limitations of the current sensor available in the market and is a considerable headway in line-following mobile robotics. Solunar's cutting-edge 16-array infrared sensor architecture, curved spatial arrangement, integrated multiplexer for easier communication, and adaptable algorithms for various situations not only improve line-following robot performance and accuracy but also open up robotics to a larger audience. The open-source compatibility of this project, encourages experiential and experimental learning fostering innovation in the field of robotics.

## 7. SUGGESTIONS FOR FUTURE WORK

Through the use of machine learning algorithms, the sensor may eventually be able to learn and adapt to different track conditions and patterns, increasing its efficacy in unpredictable situations. Developing a comprehensive library of educational resources, including tutorials and sample projects, can also encourage collaboration and help users get the most out of Solunar. Finally, looking into partnerships with educational institutions and robotics competitions could make it easier to integrate Solunar into activities and classes, which would increase its adoption and influence within the robotics community.

Solunar's curved, high-resolution 16-array infrared sensor, which includes an integrated multiplexer, improves the accuracy and user-friendliness of line-following robots, thereby increasing the accessibility of robotics. Future developments can include partnerships to broaden its use in robotics competitions and education, educational materials, and machine learning for adaptive track handling.

## References

**[1]** Baharuddin, Z., Zainal Abidin, I., Sulaiman, S., & Mohideen, K. (2005). Analysis of line sensor configuration for the advanced line follower robot. 2. Dındış, G., & Karamancıoğlu, A. (2016). An accurate line detection by a novel line follower sensor output processing method. *International Journal of Advanced Robotics Systems*, 13(1), 1-9. https://doi.org/10.1177/1729881416636144

**[2]** Abideen, Z., Anwar, B., & Tariq, H. (2018). Dual purpose Cartesian infrared sensor array based PID controlled line follower robot for medical applications. In *Proceedings of the International Conference on Electrical Engineering (ICEE)* (pp. 1-7). https://doi.org/10.1109/ICEE.2018.8566871

**[3]** *https://robu.in/ir-sensor-working/*

**[4]** *https://www.pololu.com/product/961*