Mini Project Proposal Report on

**PATH-DETECTING BOT**

****

***Submitted to***

**Department of Computer and IT Engineering**

**Everest Engineering College**

**Sanepa-2, Lalitpur**

*in partial fulfillment for the award of the degree*

*of*

**BACHELOR OF COMPUER/IT ENGINEERING**

*Submitted By*

**Ravi Shah [21075426]**

**Sakchyam Acharya [21075432]**

**Sujan Lohani [21075437]**

**Sujit Tamang [21075438]**

Date Submitted: **24/12/2023**

**ABSTRACT**

Our aim is to develop a shortest path finder bot using Arduino Mega, a popular open-source electronic controller that is widely used in the field of robotics and automation. This involves designing and implementing a software algorithm that can calculate the shortest path between two points on a given graph, and then transmitting this information to the Arduino board, which controls the bot's movements. The algorithm used in this project is based on A\* and Dijkstra's algorithm, which we learned in 3rd Semester (Data Structure and Algorithm) is a well-known algorithm for graph search that can be used to find the shortest path between two nodes in a graph. The algorithm is implemented using C++ programming language which we learned in 2nd Semester (Object Oriented Programming in C++), Arduino library and the Arduino IDE. The hardware used in this project includes an Arduino board, motor driver, IR sensors, DC motors and chassis. The system works by first detecting the starting and ending points on the graph using the sensors. Once the starting and ending points are detected, the algorithm calculates the shortest path between these two points and then transmits the instructions to the Arduino board, which controls the bot's movements. The bot then follows the instructions to navigate along the shortest path, avoiding obstacles along the way. The project is aimed at developing a low-cost and efficient solution for autonomous navigation of bots in various applications such as warehouse management, automated guided vehicles, hospital emergency direction detector and home automation. The system can be further improved by adding more sensors and using more advanced algorithms to optimize the path planning process which we tend to implement in further projects Minor (6th Semester) and Major (8th Semester).

**Table of Contents**

ABSTRACT .......................................................................................................... ii

LIST OF FIGURES ………................................................................................... iv

LIST OF TABLES ………..................................................................................... v

ABBREVIATIONS .............................................................................................. vii

Chapter 1: INTRODUCTION ............................................................................... 1

1. Background and Statement of Problem...................................................... 1
2. Objectives and scope ................................................................................. 1
3. Applications................................................................................................ 1
4. Overview of Proposal ................................................................................ 1

Chapter 2: LITERATURE REVIEW..................................................................... 2

Chapter 3: METHODOLOGY .............................................................................. 3

1. System Design ........................................................................................... 3
2. Algorithms.................................................................................................. 3
3. Hardware and software required................................................................. 3
4. Implementation Plan................................................................................... 3
5. Expected Outputs........................................................................................ 3

Chapter 4: FEASIBILITY ANALYSIS ................................................................. 4

1. Schedule...................................................................................................... 4
2. Financial Plan ............................................................................................. 4
3. Technical Feasibility................................................................................... 5
4. Operational Feasibility ............................................................................... 5

Chapter 5: CONCLUSION..................................................................................... 6

REFERENCES ....................................................................................................... 7

**LIST OF FIGURES**

Figure 3.1 Block Diagram of System ..................................................................... 3

**Important Notice / Memo Page**

**[Yo Proposal ma khi mistake haru sachyaunu parema khi thapnu parema you are free to update or delete but you have to inform or notice team member about the changes]**

**[Information and notice should be written in this page]**

1. **Notice for group member this is not the final proposal there is many mistakes and things to improve. The line or sentences which is between [] big brackets is the definition for that particular part meaning tyo thau ma chaii k lekhne vanera sir le lekheko chai ho hamro proposal hami 4 janai bata approve vaesi tyo bracket ma vaako haru sabai hatauxau until our proposal is not finished please don’t delete that part. Thank you!! From sujit**

**LIST OF TABLES**

Table 4.1 Gantt Chart .............................................................................................. 5

Table 4.2 Cost Estimation ........................................................................................ 5

**ABBREVIATIONS**

IR Sensor- Infrared Sensor

I/O- Input Output

USB- Universal Serial Bus

IDE- Integrated Development Environment

DC Motors- Direct Current Motors

RPM- Rotation Per Minute

LED- Light Emitting Diode

PCB- Printed Circuit Board

AI- Artificial Intelligence

PWM- Pulse Width Modulation

SRAM- Static Random Access Memory

KB- Kilo Bytes

EEPROM- Electronic Erasable Programmable Read Only Memory

MHz- Mega Hertz

**Chapter 1: INTRODUCTION**

**1.1 Background and Statement of Problem**

[Details on: What are the associated problems that you want to solve with this project? Also describe your project.

Problem description and justify the need for the project.]

**Background:**

In the ever-evolving landscape of robotics and automation, there is an increasing demand for intelligent systems capable of navigating and interacting with the environment autonomously. Path-Detecting Bots represent a pivotal advancement in this domain, offering the potential to revolutionize industries by providing efficient and precise navigation along predefined or dynamic paths.

Despite significant strides in robotics, the development of Path-Detecting Bot remains a complex challenge. The integration of sensors, control systems and adaptable algorithms requires a careful balance to ensure the bot’s effectiveness in diverse scenarios. The success of such a project holds promise for applications ranging from warehouse automation to smart homes and precision agriculture.

**Statement of Problem:**

The current challenges lies in creating a path-detecting bot that seamlessly combines sensor technologies, intelligent decision-making capabilities and adaptability to various environments.

**Key issues include: -**

1. **Sensor Integrations**

Identifying and integrating sensors, such as infrared or ultrasonic that provide accurate real-time data for effective path detection, obstacle avoidance and navigation.

1. **Dynamic Decision-Making**

Developing a robust control system that enables the bot to make dynamic decisions based on sensor inputs, ensuring precise navigation along predefined paths while adapting to changes in the environment.

1. **Scalability**

Designing a bot that is scalable and versatile enough to be deployed across different industries and applications, accommodating varied terrains and operational requirements.

1. **User-Friendly Configuration**

Creating an initiative user interface that allows easy configuration and adaptation of the path-detecting bot to different environments, minimize the need for specialized technical expertise.

Addressing these challenges is essential for the successful development and implementation of a Path-Detecting Bot, unlocking new possibilities for automation and contributing to increased efficiency in various sectors. This project seeks to bridge these gaps and pave the way for a transformative solution in the field of autonomous robotics.

**1.2 Objectives and Scope**

[Statements of precise and measurable outcomes. Boundary lines for the project work.]

The primary objective of a Path-Detecting Bot is to navigate autonomously in a predefined environment by detecting and following a designated path. This is achieved through sensors that gather information about the surroundings, enabling the bot to make real-time decisions on its movement to stay on the desired course.

**Some Scopes of Path-Detecting Bot are: -**

1. **Automation and Robotics**

Bots can be employed for tasks like warehouse navigation, material handling and automation in industries.

1. **Home automation**

Path-Detecting Bots are used in smart home devices, such as robotic vacuum cleaners, to navigate and clean floors efficiently.

1. **Education**

These bots serve as educational tools, helping students understand robotics, programming and sensor integration.

1. **Security and Surveillance**

Bots can be equipped with camera for patrolling and monitoring areas, enhancing security.

1. **Agriculture**

In precision agriculture, path-detecting bots can navigate fields for tasks like planting, monitoring crops, or applying fertilizers.

1. **Healthcare**

Bots may find applications in healthcare settings for tasks like transporting medical supplies within a facility.

1. **Entertainment**

Path-Detecting Bot can be used in entertainment settings, such as interactive exhibits or amusement parks.

1. **Research and Exploration**

Bots can navigate and explore environments that may be challenging or hazardous for humans, such as disaster-stricken areas or other planets.

**1.3 Applications**

[Application areas of your project.]

Path-Detecting Bots find diverse applications across various industries, contributing to automation, efficiency, and safety.

**Some notable applications include: -**

1. **Warehouse Automation**

Bots navigate through warehouses, optimizing the movement of goods, and automating tasks such as inventory management and order fulfillment.

1. **Smart Cleaning Devices**

Robotic vacuum cleaners use path detection to navigate and clean floors systematically, ensuring thorough coverage.

1. **Agricultural Robotics**

Bots equipped with path-detection capabilities are employed in precision agriculture for tasks like planting, monitoring crops, and harvesting.

1. **Manufacturing**

Bots navigate assembly lines, delivering components and ensuring the smooth flow of materials in manufacturing processes.

1. **Security and Surveillance**

Path-Detecting Bots enhance security by patrolling predefined areas, monitoring for unusual activities, and providing real-time surveillance.

1. **Healthcare Assistance**

Bots can transport medical supplies within healthcare facilities, reducing the need for human involvement in routine logistics.

1. **Educational Robotics**

Path-Detecting Bots serve as educational tools, helping students learn about robotics, programming and sensor integration.

1. **Entertainment Industry**

Bots are used in amusement parks and interactive exhibits, entertaining visitors through guided paths and programmed interactions.

1. **Research and Exploration**

Bots explore environments that may be hazardous or challenging for humans, aiding in research and exploration tasks.

1. **Delivery Services**

Autonomous delivery bots use path detection to navigate sidewalks and streets, delivering packages to specified locations.

The versatility of Path-Detecting Bots makes them valuable across a wide range of industries, enhancing efficiency, reducing labor requirements and opening up new possibilities for automation and innovation.

**1.4 Overview of Proposal**

(Not very necessary. Use this section only if your proposal or report is long)

(Note: Always page break for every new chapter)

**Chapter 2: LITERATURE REVIEW**

[Discussion and critically evaluating past and current research in your area and identify gap in the literature of your field of study that you intend to fill. Describe the methodologies that you have studied. It might be the description of the previous state of art designs similar to the project. (related papers/ projects; at least three.)]

**Literature Review: Path Following Bots**

Path following bots are robots designed to travel along specific routes without human intervention. Research has explored different ways to make these robots move accurately and safely.

**Algorithms:**

Scientists have developed various methods to help these bots follow paths. Some older techniques, like PID control, laid the groundwork. Newer methods, like reinforcement learning, use artificial intelligence to adapt better to different situations.

**Sensors:**

For these bots to navigate, they need sensors to "see" the world around them. LiDAR sensors create detailed maps, while vision-based systems using neural networks can recognize objects in real-time.

**Real-World Use:**

Path following bots are used in many industries. In warehouses, they help manage inventory efficiently. In farming, they navigate through fields, helping farmers monitor crops effectively.

**Challenges and Future:**

Despite progress, challenges remain. Bots still struggle with unexpected obstacles and rapidly changing environments. Future research might focus on combining sensor data for smarter decision-making or mimicking nature for better adaptability.

**Conclusion:**

The studies show how path following bots have evolved with smarter algorithms and advanced sensors. While they've become useful in many fields, there are still puzzles to solve, promising exciting developments ahead.

This simplified review provides a quick overview of how path following bots work, the technology behind them, their applications, challenges, and hints at future research directions.

Top of Form

**Literature review 1**

Path following bots are like smart robots that follow a set path by themselves. Scientists use different clever ways to make them move correctly. They use special sensors, like eyes for robots, to understand the world around them. These bots help in warehouses and on farms, making jobs easier for people. Even though they're super helpful, there are still some tricky problems to solve, like dealing with surprises in their way. Scientists are working hard to make them even smarter for the future.

Top of Form

**Literature review 2**

Scientists have been making cool robots called path following bots that can move on their own along specific paths. They use smart tricks to help these robots follow the path just right. To see where they're going, these bots have special sensors that act like their eyes. People use these bots in places like warehouses and farms to do important jobs. Even though these robots are really helpful, scientists are still trying to make them even better at handling unexpected things that might pop up in their way.

Top of Form

**Chapter 3: METHODOLOGY**

[Describe the approach and methodologies used to address the problem.]

**Methodology for Path-Detecting Bot:**

1. **Requirements Analysis**

* Identify the specific applications and environmental conditions for the Path-Detecting Bot.
* Define the required features, such as sensor types, navigation precision and adaptability to different terrains.

1. **Sensor Selection and Integration**

* Evaluate sensor technologies (i.e. Infrared, Ultrasonic) based on accuracy, range and environmental suitability.
* Integrate selected sensors to provide real-time data for path detection and obstacle avoidance.

1. **Hardware Designs**

* Develop a scalable and adaptable hardware architecture that accommodates the selected sensors, control systems and power supply.
* Optimize the design for size, weight and power consumption based on the targeted application.

1. **Control System Development**

* Implement control algorithms for autonomous navigation, considering dynamic decision making and obstacle handling.
* Fine-tune the control system to ensure the bot follows predefined paths accurately and efficiently.

1. **Software Implementation**

* Develop software modules for sensor data processing, path planning and motor control.
* Integrate a user-friendly interface for easy configuration and customization of the bot’s behavior.

1. **Testing in Controlled Environments**

* Conduct initial tests in controlled environments to validate sensor accuracy, path-following precision and obstacle avoidance capabilities.
* Refine algorithm and system parameters based on test results.

1. **Optimization and Iteration**

* Optimize the bot’s performance based on testing feedback.
* Iterate on hardware and software components to enhance reliability, efficiency and adaptability.

1. **Documentation**

* Document the hardware specifications, control algorithms and software architecture.
* Provide user manuals and guidelines for configuring and maintaining the Path-Detecting Bot.

1. **Deployment**

* Prepare the Path-Detecting Bot for developing in the targeted application.
* Provide training and support for end-users, ensuring effective integration into their operational processes.

This comprehensive methodology ensures a systematic approach to the development of a path-detecting bot, covering all aspects from initial requirements analysis to deployment and user training. Regular testing and iteration are essential to refining the bot’s capabilities and ensuring its success in real-world scenarios.

**3.1 System Design**

[Go for a high level block diagram. This section may includes:

Block Diagram, Use Case, DFDs, Sequence, ER-diagrams, State diagrams etc.

No need to include all above diagram.

(You may describe the then after.)]

**3.2 Algorithms**

[Describe the algorithms that you will be implementing to solve the problem that you have stated.]

**3.3 Assumptions**

[Enumerate the assumptions that you have made for the project.]

To develop a Path-Detecting Bot, its essential to make certain assumptions, such as:-

1. **Input Source**

Assuming the bot receives input from a digital platform or communication channel.

1. **Project Structure**

Assuming projects follow structured format or have identifiable patterns in their paths.

1. **File Systems**

Assuming the bot interacts with file systems or repositories to trace project paths.

1. **User Interaction**

Assuming the bot may need to prompt users for additional information or clarification during the path detecting process.

1. **Common Conventions**

Assuming there are common naming conventions or markers that help identify project paths.

1. **Security and Permissions**

Assuming the bot has the necessary permissions to access and analyze project paths.

1. **Dynamic Environments**

Considering the possibility that project paths can change over time due to updates or modifications.

1. **Error Handling**

Assuming the bot includes error-handling mechanisms to address cases where project paths are ambiguous or inaccessible.

**3.4 Hardware and software required**

[Describe your hardware and software that you will be needing in brief.

You may include:

* Basic features of the component/software.
* Define the reason why you choose these component/software]

**Hardware Components required for Path-Detecting Bot are:**

1. **Arduino (Mega)**

The Arduino Mega is a microcontroller board based on the ATmega2560.

Brief summary of the Arduino Mega:

1. **Microcontroller:** The Arduino Mega is powered by the ATmega2560 microcontroller, offering more I/O pins and memory compared to standard Arduino boards.
2. **I/O Pins:** It features a total of 54 digital input/output pins, with 15 providing PWM (Pulse Width Modulation) output.
3. **Analog Inputs:** The board includes 16 analog input pins, allowing for a wide range of sensor connections.
4. **Memory:** With 256KB (Kilo Bytes) of flash memory for storing the program, 8KB of SRAM (Static Random Access Memory), and 4KB of EEPROM (Electronic Erasable Programmable Read Only Memory), the Mega provides ample space for complex projects.
5. **Clock Speed:** Operates at a clock speed of 16 MHz(Mega Hertz), providing sufficient processing power for various applications.
6. **USB Connection:** Equipped with a USB interface for easy programming and serial communication with a computer.
7. **Power Supply:** The board can be powered via USB connection or an external power supply (7-12V).
8. **Compatibility:** Arduino Mega is compatible with most Arduino shields, allowing for easy expansion and compatibility with a wide range of accessories.
9. **Voltage Regulator:** Include a voltage regulator, enabling a stable power supply to the microcontroller and other connected components.
10. **Versatility:** Designed for larger and more complex projects due to its extensive I/O capabilities and memory, making it suitable for robotics, automation and various electronic applications.
11. **Motor Control (Motor Driver L293D)**
12. **Motors and Wheels**
13. **Sensors (Infrared and Ultra Sonic)**
14. **Power Supply (Rechargeable Battery)**
15. **Chassis**
16. **Wires and Connectors (Jumper Wire)**
17. **Breadboard**
18. **Encoders**

**Software Components required for Path-Detecting Bot: -**

**3.5 Implementation Plan**

[(How are you going to implement the system? Eg. programming language you choose, why are going to choose that particular language)]

**Simplified Implementation plan for Path-Detecting Bot: -**

1. **Define Requirements:** Make proper list and outline of bot’s objective and functionalities. Properly specify the type of path detection.
2. **Select Components:** Get all required items for the implementation of project. i.e. Arduino, Chassis, Motor, Wheels, Batteries, Sensors, etc.
3. **Create Circuit:** Set up circuit on a breadboard initially for prototyping. Connect motors, sensors and other components to the Arduino Mega using wires.
4. **Write and Upload Code:** Develop Arduino Code using C++ to control motor movement and process sensor input. Implement logic for path detecting and navigation using Dijkstra's Algorithm and A\* Algorithm. Upload the code to the Arduino Mega using Arduino IDE (Integrated Development Environment).
5. **Test and Debug:** Test the bot in controlled environment to ensure it follows the path correctly and smoothly. Debug and refine the code to address any issues.
6. **Build the Chassis:** Construct or assemble the physical structure (Chassis) to hold the Arduino and other components securely.
7. **Integrate Components:** Mount motors, wheels and sensors onto the chassis. Ensure proper wiring and connectivity.
8. **Power Supply:** Connect Battery to power the bot.
9. **Fine-Tune Sensors:** Adjust sensors positions and calibrate their sensitivity for optimal path detection.
10. **Refined and Optimization:** Fine-tune the bot’s movement and path-following algorithms. Optimize code for efficiency and responsiveness.
11. **Documentation:** Document the circuit diagram, code and any modification made during the implementation.
12. **Testing in Real Environment:** Test the bot in the actual environment for the project.
13. **Iterate and Improve:** Gather feedback from testing and make necessary iterations for improvement.

**3.6 Expected Outputs**

[Expected output or deliverables of your project work.]

**Some expected outputs for a Path-Detecting Bot are: -**

1. **Motor Control:** The bot should demonstrate controlled movement, adjusting its motors to follow a predefined path.
2. **Path Following:** Successful navigation along a specified path, with the ability to detect and respond to going off course.
3. **Sensor Feedback:** Output from sensors indicating the bot’s perception of the path, such as sensor readings or activation signa ls.
4. **Status Indicators:** LEDs or other indicators showing the bot’s operational status, such as whether it is actively following the path or if there are errors.
5. **Serial Communication:** Serial communication output to a connected device (i.e. Laptop) for monitoring or debugging purposes.
6. **Power and Battery Status:** Indicators or Feedback on the remaining battery life or power status.

These outputs collectively contribute to the bot’s functionality, allowing it to follow paths, respond to environmental changes and provide feedback on its operational state.

**Chapter 4: FEASIBILITY ANALYSIS**

[State why your project is feasible in terms of economy, time, technical and operational etc.]

Creating a path-Detecting Bot using Arduino is feasible. Arduino offers a versatile platform for hardware projects and we can integrate sensors like Infrared or Ultrasonic sensors for path detection.

**4.1 Schedule**

[This section defines the feasibility of the project in terms of time. Estimate the duration of your project, and identify order in which activities should be performed.]

**Table 4.1 Gantt Chart**

**4.2 Financial Plan**

[This section defines the feasibility of the project in terms of economy. A line-item budget to support the project costs, including any cost sharing or program income.]

**Table 4.2 Cost Estimation**

**Materials Required For This Project With Their Approx. Cost Estimation:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.N** | **Name of Materials** | **Price per piece (in Rs.)** | **Required Quantity** | **Total Price (in Rs.)** |
| 1. | Arduino Mega | 2000 | 1 | **2000** |
| 2. | Motor Driver L298 | 350 | 1 | **350** |
| 3. | Ultrasonic Sensor | 180 | 1 | **180** |
| 4. | Infrared Sensor | 95 | 4 | **380** |
| 5. | Jumper Wire | 5 | 50 | **250** |
| 6. | LDR | 15 | 5 | **75** |
| 7. | 4 Wheel Set Chassis | 1300 | 1 | **1300** |
| 8. | Breadboard | 160 | 1 | **160** |
| 9. | Rechargeable Battery | 780 (Per 2 Batteries) | 8 | **3120** |
| 10. | Battery Charger | 780 | 1 | **780** |
| 11. | Servo Motor | 230 | 3 | **690** |
| 12. | Resistor |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
| **Total** |  |  |  | **9,535(Approx.10,000)** |

**So, Final Expected Budget from College Department=Rs 10,000**

**4.3 Technical Feasibility**

[(Require to evaluate viability of your project system architecture and design in sense of technology you used.)]

A project Path-Detecting Bot is technically feasible using computer vision, machine learning or a combination. Further Implementation in the future project would require image processing algorithms to analyze paths, obstacle detection and navigation systems for effective implementation. The feasibility depends on specific requirements and environmental factors.

**4.4 Operational Feasibility**

[(Require to evaluate whether your team is able to complete this project under all above defined criteria. Also be sure after completion of project, what would be the requirement to Maintain and Operate. (They may be costing, additional resources, legal requirements etc.))]

**Chapter 5: CONCLUSION**

[Description of expected benefits of the proposed project and re-emphasize context and value of proposed project.]

The development of a Path-Detecting Bot using Arduino provides a cost-effective and efficient solution for autonomous navigation of bot. By using Arduino, the system can be easily customized and modified according to specific requirements. However, the interaction with sensor has some limitations. Because due to intensity of environmental lighting effect leads to error in sensor signal. More over the battery should be recharged frequently making it suitable for various applications such as warehouse management, automated guided vehicles, and home automation. However, the system can be further improved by incorporating more advanced algorithms and adding more sensors to optimize the path planning process. Overall, the project provides a solid foundation for the development of autonomous navigation systems using Arduino and can serve as a starting point for more advanced robotics applications.

**REFERENCES**

[1] Keller, T. E., Cusick, G. R., & Courtney, M. E. (2007). Approaching the transition to adulthood: Distinctive profiles of adolescents aging out of the child welfare system. Social Services Review, 81, 453-484.

[2] Alexie, S. (1992). The business of fancydancing: Stories and poems. Brooklyn, NY:

Hang Loose Press.

Follow IEEE Citation Reference format.

<http://www.ieee.org/documents/ieeecitationref.pdf>

**Bibliography**

Note: All the chapters are mandatory, but the subheadings can be used as per your requirement.