INTERNATIONAL JOURNAL OF PROGRESSIVE RESEARCH IN ENGINEERING MANAGEMENT AND SCIENCE (IJPREMS)

www.ijprems.com
editor@ijprems.com

Vol. 03, Issue 06, June 2023, pp : 145-147

e-ISSN : 2583-1062

Impact Factor : 5.725

SMART CITY TRAVELER USING IOT

Mahender kumar[1], Dr. Santosh kumar Dwivedi[2], Mr. Aakash Srivastava[3]

[1]UG Student of Department of Bachelor of computer Application, Shri Ramswaroop Memorial college of management Lucknow, Uttar Pradesh, Indian.

[2]Professoor, Head of Department of Bachelor of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

[3]Assistant professor, Department of Bachelor of Computer Application, Shri Ramswaroop Memorial College of Management Lucknow, Uttar Pradesh, India.

ABSTRACT

The abstract of the book "Smart City Traveler Using IoT" describes a concept that uses the Internet of Things (IoT) to improve the travel experience in smart cities. With the increasing connectivity of urban environments, the integration of IoT technologies enables various devices and systems to communicate and collaborate effectively.

In this context, the smart city traveler concept focuses on using the Internet of Things to create intelligent and seamless travel for individuals within the infrastructure of a smart city. Utilizing IoT devices, sensors and data analytics, the concept aims to solve key issues facing travelers such as traffic congestion, navigation, safety and access to relevant information.

1. INTRODUCTION

In recent years, the concept of smart cities has gained significant attention as urban areas seek to use technology to improve the quality of life for their residents. The Internet of Things (IoT) has emerged as a transformative force in the field of smart cities, enabling the integration of various devices, systems and data to create a more efficient and sustainable urban environment.

One of the key areas where the Internet of Things can have a profound impact is in travel and transportation. As cities become more populated and congested, there is a growing need to develop innovative solutions that optimize the travel experience for residents and visitors alike. The "Smart City Traveler Using IoT" concept aims to address these challenges by using IoT technologies to create a seamless and intelligent travel ecosystem within smart cities. The primary goal of the smart city traveler concept is to improve the overall travel experience by leveraging connectivity, real-time data and advanced analytics. By deploying IoT-enabled devices and sensors throughout the city, valuable information can be collected and analyzed to provide accurate and timely information to travelers

2. WORKFLOW

The workflow for a line tracking and obstacle avoidance project can be broken down into several steps. Here is a general overview of the workflow:

Design and Assembly: Design the chassis of the robot and assemble the necessary components such as motors, wheels, sensors and a microcontroller or development board (eg Arduino or Raspberry Pi) to control the robot.

Line Tracking Algorithm: Implement a line tracking algorithm that allows the robot to detect and track a line on the ground. This can be achieved using sensors such as infrared (IR) or color sensors that detect the position of the line relative to the robot.

Obstacle detection: Include obstacle detection sensors such as ultrasonic sensors, infrared distance sensors, or cameras that detect objects or obstacles in the robot's path.

3. PROPOSED SYSTEM

Hardware components: Robotic chassis with motors and wheels Microcontroller or development board (e.g. Arduino or Raspberry Pi) Line sensor (e.g. infrared or color sensor) for line detection Obstacle detection sensors (e.g. ultrasonic sensors or infrared distance sensors) to detect obstacles. A motor controller or motor control circuits to control the movement of the robot.

Line Tracking Algorithm: Calibrate the line sensor to detect the line on the ground. Read the sensor values and determine the position of the line relative to the robot. Use a Proportional Integral Derivative (PID) control algorithm to adjust the robot's motor speed based on the position of the line, allowing it to precisely follow the line.

Obstacle Avoidance Algorithm: Read the sensor values of the obstacle detection sensors to determine the presence and distance of obstacles. Implement a decision algorithm that determines the appropriate action based on the location and distance of the obstacle. If an obstacle is detected, the robot should stop or change direction to avoid the obstacle. After avoiding the obstacle, the robot should continue following the line.

Integration: Combine line tracking and obstacle avoidance algorithms to create a cohesive system. Implement logic that allows the robot to switch between line following mode and obstacle avoidance mode based on sensor inputs. Ensure smooth transitions between the two modes without compromising the functionality of the robot.

4. SYSTEM OVERVIEW

Hardware components: Robotic Chassis or Vehicle Platform: Provides the structural basis for the project. Motor Controllers: Controls the movement of the robot's wheels or motors.

Line Detection Sensors: Typically, infrared (IR) sensors or cameras are used to detect lines on the ground.

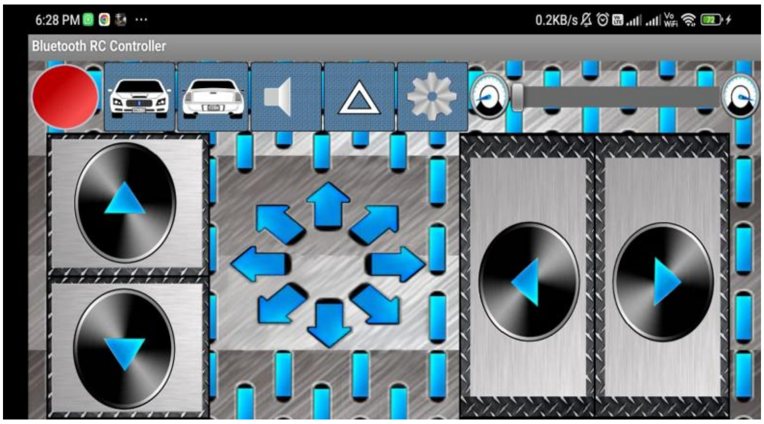
Obstacle detection sensors: Ultrasonic sensors, infrared sensors or cameras can be used to detect obstacles. Microcontroller or Single Board Computer (SBC): Serves as the brain of the robot and controls all the components.

Power: Battery or external power source to power the robot.

Software components: Operating System or Real-time Operating System (RTOS): Provides a platform for the robot software to run.

Motor Control Algorithms: These algorithms control the movement of the robot based on the input from the guidance detection sensors.

Line Tracking Algorithm: Processes the sensor data and calculates the necessary motor control signals to keep the robot on the line.



5. ANALYSIS

After the line: Line tracking involves designing a robot to detect and follow a line (usually of contrasting color) on the ground using sensors such as infrared or color sensors. A line usually represents a pre-defined path that the robot must follow. Here are the key steps involved in the following line:

Sensor Data Collection: The robot collects sensor data to determine the position and orientation of the line relative to its current position.

Line Detection: The acquired sensor data is processed to identify the line. Techniques such as thresholding, filtering or edge detection can be used for line detection.

Error calculation: The robot calculates the error by comparing its position with the desired position on the line. This error represents how far the robot deviates from the desired path.

Control Action: Based on the calculated error, the robot will use control algorithms such as PID (Proportional Integral Derivative) to adjust its motion and head towards the line. A control action may include adjusting the engine speed or changing the direction of the steering mechanism.

6. CONCLUSION

In conclusion, the obstacle avoidance system in a track-following car is a significant advance in the field of autonomous driving and vehicle automation. It combines the principles of line detection and obstacle avoidance to allow the car to autonomously follow a predetermined path while avoiding obstacles in its path. The system uses various sensors such as cameras, ultrasonic sensors or RIDAR to perceive the environment and make informed decisions. The implementation of such a system offers several advantages, including increased safety, greater efficiency and reduced driver workload. By autonomously navigating the road and avoiding obstacles, it can help prevent accidents and collisions. The system could be particularly useful in applications such as warehouse automation, industrial environments or even self-driving cars in smart cities.

7. FUTURE SCOPE

Seamless multimodal transportation: Apps for smart urban travelers can use IoT to enable seamless integration of multiple transportation modes, including public transportation, ride-sharing services, bike sharing, and autonomous vehicles. By aggregating and analyzing data from different transport systems in real time, these applications can offer passengers optimized routes and options for efficient, comfortable and environmentally friendly journeys.

Real-time Traffic Management: IoT-enabled smart city traveler apps can play a vital role in real-time traffic management. By collecting data from connected vehicles, traffic sensors and infrastructure, these applications can provide accurate and up-to-date information on traffic conditions, congestion and traffic accidents. This data can be used to optimize traffic flow, design alternative routes and improve overall traffic efficiency.

ACKNOWLEDGEMENT

The Conclusion of any task is a reward not only for those who enthusiastically engage in it, but also for the people involved in inspiring, controlling and helping these nations. I would like to thank everyone here who helped me complete this project without which it would have been a colossal task indeed. However, this project would not have been possible without the ruthlessness, care and support of many people. I would like to express my deepest gratitude to everyone who made it possible for me to complete this project report. Special appreciation to my final year project guide, Mr. Aakash Srivastava, whose contribution is support of suggestions and encouragement helped me align my project, especially while writing this report.Special thanks to all the faculty members of SRMCM who have been giving me great support in the development of this project. I also thank all my friends who from time to time gave me recommendations and inspiration to work on this project.

8. REFERENCES

[1] Pearson, Software Engineering, Tenth Edition, 2008.

[2] Roger S. Pressman Software Engineering, 3rd edition by McGraw Hill International, April 2002.

[3] A. K. Majmudar P. Bhatta charya, Database Management System, TMH,Third Edition, 199.