Cyclite- Smart Cycle Indicator

Rupali Gavraskar, Mrinal Potdar, Prerak Gadpayale, Khushi Purwar, Akshara Rakhe, Raghavi Narayanan

Department of Electronics and Telecommunication (ENTC)
Vishwakarma Institute of Technology, Pune, 411037, Maharashtra, India

Abstract

This study introduces a brand-new gesture-controlled bicycle indication system that improves rider safety by using clear signaling. Two Arduino Nano microcontrollers, an MPU6050 inertial measurement unit (IMU), and HC-05 Bluetooth modules are used in the master-slave setup of the system. The MPU6050 is used by the master unit, which is built into the cyclist's glove, to record hand tilt movements. The slave device mounted on the bicycle receives these gestures wirelessly. To indicate left or right turns, the slave unit decodes the data and turns on the appropriate LED indicators. By using this hands-free method, bikers can signal while keeping control, increasing visibility, and lowering the chance of an accident. The system's performance, design, and implementation are described in detail, emphasizing how it could enhance both user experience and cycling safety.
Keywords *— Arduino Nano, Arduino IDE,* MPU-6050, HC-05 Bluetooth module, Master and Slave, Gesture control

I. INTRODUCTION

As a sustainable, healthful, and effective form of transportation, cycling has seen a global renaissance, helping to improve public health, reduce traffic congestion, and cut carbon emissions. But the growing number of bikers on the road has also raised questions about their safety, especially in situations where there is shared traffic. Due to their inherent vulnerability, cyclists are more likely to be involved in accidents than drivers, and many of these incidents are caused by insufficient or ambiguous intention signals, particularly when turning. Although required in many jurisdictions, traditional hand signals force cyclists to take their hands off their handlebars, which temporarily impairs their balance, steering control, and general stability. This becomes more dangerous in difficult situations, including slippery or rainy roads, rough terrain, or when transporting heavy objects. Additionally, hand gestures may not always be seen, particularly at night or in bad weather, which raises the possibility of crashes with other cars or pedestrians.

In order to reduce these risks and give bikers a safer and more efficient way to communicate their intentions, creative solutions are required. In order to improve cycling safety through simple, hands-free signaling, this research presents a unique gesture-controlled indication system. The system consists of a wearable glove with a wireless communication module and an inertial measurement unit (IMU). Hand signals that indicate left and right turns are picked up by the IMU and wirelessly sent to a receiver unit attached to the bicycle. The receiver clearly indicates the cyclist's intended direction to oncoming traffic by turning on matching high-visibility LED indicators when it receives the signal. By using this method, bikers can continue to control their bike while signaling, increasing stability and lowering the risk of collisions.

The design, development, and performance assessment of the gesture-controlled indication system are described in depth in this study. The development of gesture recognition algorithms, the implementation of wireless communication protocols, the choice of hardware components, and the incorporation of the system into a working prototype will all be covered.

II. LITERATURE REVIEW

rafiqul zaman khanand ,noor adnan ibraheem[1]hand gesture recognition systems aim to create natural human-computer interaction by enabling computers to understand and interpret hand gestures. the process involves three main steps: segmenting the hand from the input image, extracting relevant features, and classifying the gesture using algorithms like neural networks or hidden markov models. common techniques include using skin color for segmentation and various shape-based features. these systems have diverse applications, including sign language translation, robot control, and virtual environments. k.t.krishna kumar, medisetti abimitra[2]the literature review within the provided document explores the technological foundations, challenges, applications, and future directions of gesture-based cursor control systems as a transformative paradigm in human-computer interaction (hci). early systems, such as microsoft kinect and leap motion, pioneered gesture recognition using depth sensors, enabling applications in gaming and virtual environments. these systems employ algorithms for hand detection, feature extraction, and gesture classification, with advancements now integrating machine learning to improve accuracy and robustness. commercial implementations like samsung's air gesture and google's project soli have demonstrated successes in gaming, interactive displays, and automotive interfaces, showcasing improved user engagement and accessibility for individuals with disabilities. TWINKLE SHUKLA, PRADIP MAHAJAN[3]this system uses a webcam to capture hand gestures via color detection. users wear colored caps or tapes on their hands, allowing them to navigate the cursor by moving their hands. the system recognizes different hand gestures to perform various cursor functions, such as right and left clicks, double clicks, and scrolling. it relies solely on a low-resolution webcam as a sensor, tracking the user's hand in two dimensions and interpreting up to five distinct hand gestures. the system is implemented in matlab using the image processing toolbox and opencv kevin yeh (kky226)[4]cursor control using hand gestures leverages computer vision and machine learning to interpret hand movements, enabling users to manipulate a cursor without physical contact. systems often use webcams and color detection to track hands, recognizing gestures that correspond to actions like movement, clicking, and scrolling. while effective, challenges include environmental sensitivity, latency, and achieving high accuracy. the literature also highlights the use of accelerometer and gyroscope data from wearable devices for gesture recognition. these systems aim to provide intuitive, hands-free interfaces for diverse applications, though current implementations still strive to balance accuracy, responsiveness, and user adaptation. karthik ramani, sang ho yoon, yanke tan[5]cursor control systems using hand gestures, as explored in "cursor control using hand gestures" and "bikegesture," employ computer vision, machine learning, and wearable sensors to interpret hand movements for various functions. "cursor control using hand gestures" focuses on webcam-based systems using color detection and algorithms like opencv, while "bikegesture" emphasizes subtle microgestures captured through wearable sensors for cycling-related inputs. both approaches aim to provide intuitive interfaces, but face challenges in environmental sensitivity, accuracy, and latency. user preferences lean towards subtle, easily executable gestures that ensure safety and efficient control, with prototypes demonstrating promising accuracy and response times compared to traditional mechanical inputs. alexandru dancu, velko vechev[**6**]based on the documents "cursor control using hand gestures," "bikegesture: user elicitation and performance of micro hand gesture as input for cycling," and "gesture bike: examining projection surfaces and turn signal systems for urban cycling," gesture recognition systems for cycling prioritize intuitive and safe interfaces. "bikegesture" highlights a user preference for subtle microgestures to maintain control and safety, achieving 92% accuracy in its prototype, while "gesture bike" found that gesture-enabled turn signals were easier to use than commercial systems. both documents consider that gesture controls should align with user expectations and existing practices; simplicity and ease of learning are crucial, and systems must effectively balance accuracy, responsiveness, and minimizing distractions to ensure cyclist safety and usability.

III. METHODOLOGY

Components	Description
Arduino Nano (x2)	Microcontroller for both modules
MPU6050	6-axis IMU for detecting hand gestures
HC-05 Bluetooth Module (x2)	Wireless communication between modules
LEDs (x2)	Visual indicators for left and right turn
Resistors	Voltage divider for safe serial communication

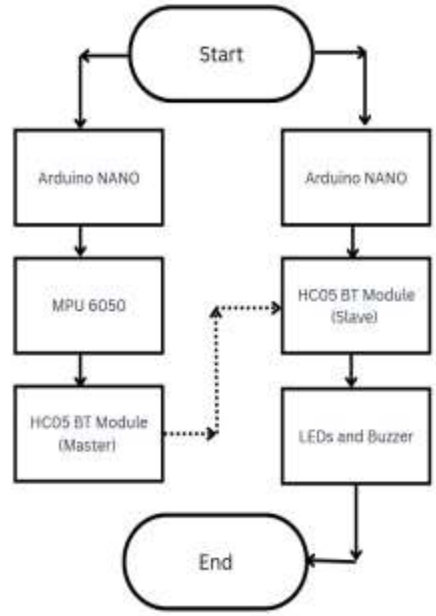

Figure: Flowchart

This section outlines the systematic approach adopted for developing a wireless, gesture-controlled indicator system for cyclists. The methodology includes hardware design, sensor integration, wireless communication setup, and software implementation using embedded programming.

I. System Architecture

1) The system is divided into two modules:

- Master Module: Worn on the user's wrist, this includes an Arduino Nano, an MPU6050 accelerometer and gyroscope sensor, and an HC-05 Bluetooth module configured in master mode.
- Slave Module: Mounted on the bicycle, it includes another Arduino Nano, an HC-05 Bluetooth module (configured as slave), and two LEDs representing left and right indicators.

Wireless communication is established between the master and slave modules to transmit gesture commands from the rider to the bike-mounted indicators..

a) Hardware Components

A **voltage divider** using 2.2kΩ and 1kΩ resistors was used between the Arduino TX and HC-05 RX pins to step down the 5V logic to 3.3V, ensuring safe operation of the Bluetooth modules.

Table 1: Components and Description

Transmission

The MPU6050 sensor captures real-time acceleration values along the X-axis to detect wrist tilts. These values are mapped to a 0–255 range and thresholds are defined to interpret gestures:

- **Left tilt**: X-axis value < 40 → “LEFT”
- **Right tilt**: X-axis value > 220 → “RIGHT”

The master module continuously reads sensor data and sends the appropriate gesture command via Bluetooth if a gesture is detected. A delay of 300 ms is used to reduce noise and accidental readings.

c) Bluetooth Configuration

The HC-05 modules were configured using **AT commands**:

- **Master Module**:
  - `AT+ROLE=1` → Set as master
  - `AT+CMODE=0` → Connect to specific address
  - `AT+BIND=XXXX,XX,XXXXXX` → Bind to slave’s MAC address

- **Slave Module**:
  - `AT+ROLE=0` → Set as slave
  - After configuration, the master automatically connects to the slave on start-up.

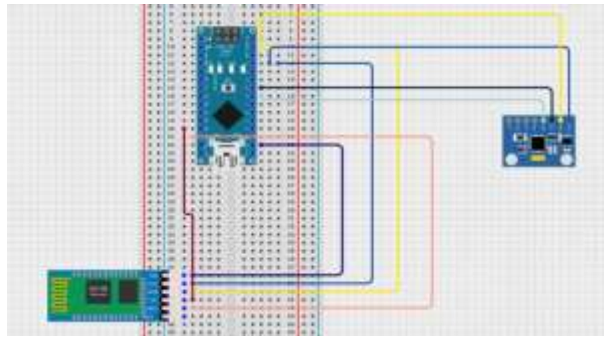

Figure: Master Module Circuit

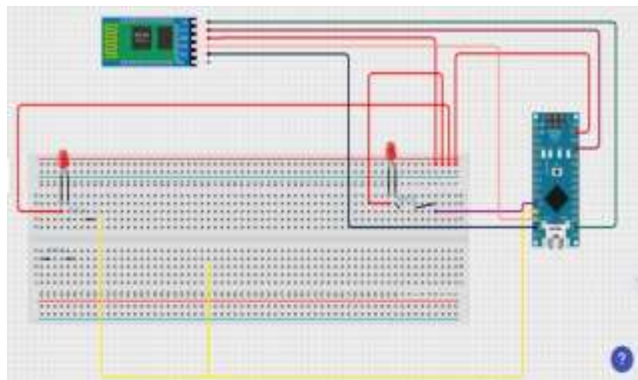

Figure: Slave Module Circuit

d) LED Signaling Logic

The slave module receives the "LEFT" or "RIGHT" command via Bluetooth. Based on the command:

- The corresponding LED is toggled ON/OFF every 500 ms (blinking).
- The LED blinks for 10 seconds and then turns off.
- The other LED is ensured to be off during activation to avoid confusion.

2) Software Implementation

The code for both modules was developed using the Arduino IDE in C. The Software-Serial library was used for Bluetooth communication. The master code focuses on gesture detection and command transmission, while the slave code manages Bluetooth input and LED blinking logic.

3) Testing Procedure

- Bluetooth communication was tested using serial monitors and echo codes.
- Gesture recognition was validated by observing MPU6050 output values and checking for consistent mapping to expected gestures.
- LED signal was tested in response to valid gesture commands.

Testing scenarios included various tilt angles, shaking noise, and quick directional changes to check system responsiveness and reliability.

IV. Results and Discussions

Figure: Left Tilt

Figure: Right Tilt

The gesture-controlled indicator system was successfully implemented and tested on a bicycle using two Arduino Nano boards, an MPU6050 sensor, and HC-05 Bluetooth modules. The system operated wirelessly, with one Arduino (master) mounted on a glove worn by the cyclist and the second Arduino (slave) attached to the rear of the bicycle controlling LED indicators.

- When the cyclist tilts their hand to the left, the system correctly interprets this as a left-turn signal. The slave module activates the green LEDs on the left side of the indicator board, as shown.
- When right-hand tilt is detected and processed. The system responds by illuminating the green LEDs on the right side of the indicator board, confirming accurate communication and response.
- The LEDs were clearly visible during night-time testing, ensuring high visibility and safety.

Discussions

The primary objective of this study was to design and implement a gesture-controlled indicator system for cyclists using wireless communication between two Arduino Nano modules connected via HC-05 Bluetooth devices. The results demonstrate that a cost-effective, lightweight, and wearable solution is feasible for improving cyclist safety by enabling non-verbal, hands-free signaling. The system effectively utilized the MPU6050 inertial measurement unit (IMU) to detect specific hand gestures, which were processed on the master module and transmitted wirelessly to the slave module responsible for LED signaling. The communication between the modules was successfully established through AT command configuration of the HC-05 modules in master-slave mode. Although initial pairing and serial communication setup posed challenges, including unstable connections and inconsistent baud rate settings, these were resolved through appropriate use of AT commands and implementation of a voltage divider on the HC-05 RX line to ensure compatibility with 3.3V logic levels.

V. FUTURE SCOPE

Seamless Smart Device Integration: Rather than requiring a dedicated glove, future iterations should focus on integrating the gesture recognition technology directly into commonly used smart devices like smartwatches, fitness trackers, or even cycling computers. This approach would significantly enhance user convenience by eliminating the need for an additional wearable device, streamlining the user experience and increasing the likelihood of adoption. The system could leverage the existing sensors and processing capabilities of these devices, reducing the overall cost and complexity.

1) Advanced and Context-Aware Gesture Recognition: Implement more sophisticated gesture recognition algorithms that go beyond simple tilt detection. This could involve utilizing machine learning techniques to recognize a wider range of gestures, adapt to individual user styles, and improve accuracy in varying conditions.
2) Haptic Feedback for User Confirmation and Enhanced Safety: To provide users with clear and immediate feedback on the system's operation, incorporate haptic feedback into the wearable device. This could involve using a small vibration motor to provide a tactile confirmation when a gesture is recognized and the corresponding signal is activated. The intensity and pattern of the vibration could be customized to indicate different actions or alerts.
3) Vehicle-to-Vehicle (V2V) Communication for Direct Signal Transmission to Drivers: Incorporate V2V communication capabilities to directly transmit turn signal information to nearby vehicles. This would provide an additional layer of safety by alerting drivers to the cyclist's intentions, even if they are not directly visible. The system could use standardized V2V communication protocols to ensure interoperability with other vehicles on the road.
4) Optimized Power Efficiency for Extended Battery Life: Optimize the power consumption of the system to extend battery life and reduce the need for frequent charging. This could involve using low-power components, implementing efficient wireless communication protocols, and incorporating power-saving modes that automatically disable certain features when they are not needed. The system should also provide users with clear indications of battery life and charging status.

VI. Conclusion

This paper presented a gesture-controlled indicator system for bicycles designed to enhance cyclist safety by providing a hands-free method of signaling. The system, utilizing an Arduino Nano, MPU6050, and HC-05 Bluetooth modules in a master-slave configuration, successfully translates hand gestures into clear and visible LED turn signals. By allowing cyclists to maintain control of their handlebars while signaling, this system addresses a critical safety concern associated with traditional hand signals. The results of our experiments demonstrate the feasibility and effectiveness of the proposed system. Future work will focus on refining the gesture recognition algorithms, integrating the system with smart devices, and exploring advanced features such as GPS-based automation and V2V communication to further improve cycling safety and user experience.

VII . Acknowledgment

The authors would like to express sincere thanks to Management and Principal of Vishwakarma Institute of Technology, Pune, India, Head of the Department of Electronics and telecommunication Engineering, and all those who supported us directly and indirectly during the project.

References

[1]” Electrical and Electronics Engineering Department, Hirasugar Institute of Technology, Nidasoshi, Karnataka, India” Design and Implementation of Gesture Controlled Bicycle Indicator Gloves

[2] Yanke Tan, Sang Ho Yoon, Karthik Ramani, BikeGesture: User Elicitation and Performance of Micro Hand Gesture as Input for Cycling May 6–11, 2017, Denver, CO, USA

[3]. “Twinkle Shukla1, Pradip Mahajan2, Ayush mandlik3, Pradip Divekar4, Suraj Burange5 “Cursor Control Using Hand Gestures

[4] “Kevin Yeh (kky226) CS 386W: Wireless Networking Fall 2014 | Austin, TX” LiteTurn: Automated, Gesture-Controlled Cyclist Turn Lights Using Cheap and Efficient Consumer Devices

[5]” Philip Jonah Ediguah” Hand Gesture Recognition and Control for Human-Robot Interaction Using Deep Learning

[6] “Mr. Chandan R P1 , Basavaraj P Battikoppa2 , Mr. Gurukiran Nayaka” Design and Implementation of Smart Cycle indicatorVolume 3, Issue 10, May 2023