IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

AI VOICE CONTROLLED HEXAPOD ROBOT

[1]Bodigi Sneha, [2]Galla Satya Girinadh, [3]Ch.Hema Latha, [4]Budda Raju Satya Surya, [5]Ippili Bhargav Kumar, [6]Mrs. M. Padma Priyanka

[1,2,3,4,5,]Student, [6]Assistant Professor

[1]Electronics And Communication Engineering,

[1]Nadimpalli Satyanarayana Raju Institute Of Technology, Visakhapatnam, India

Abstract: This paper presents the design and implementation of an AI-driven voice-controlled hexapod integrating natural language processing and real-time control for enhanced human-robot interaction. The hexapod robot, with its six-legged locomotion, offers superior stability and mobility over uneven terrain, making it ideal for exploratory, rescue, and surveillance applications. The system utilizes a speech recognition module powered by deep learning models to accurately interpret voice commands and translate them into coordinated robotic movements via inverse kinematics algorithms. An onboard microcontroller manages real-time actuation, while wireless communication allows seamless remote control and monitoring. Experimental results demonstrate the robot&#39;s capability to respond to a variety of natural voice commands with high accuracy and minimal latency, showcasing its potential in both indoor and outdoor autonomous operations. The integration of AI voice interfaces with multi-legged robotics opens up new possibilities in intuitive, hands-free control mechanisms forcomplex robotic platforms.

Index Terms **-Hexapod Robot, Voice Control, Artificial-Intelligence, Speech Recognition, Inverse Kinematics, Real-Time Control, Human-Robot Interaction, Mobile Robotics.**

I. INTRODUCTION

Hexapod robots offer excellent stability and mobility on uneven terrain, making them suitable for rescue, exploration, and surveillance tasks. Meanwhile, AI-powered voice control enables intuitive, hands-free interaction, which is especially valuable in critical or hazardous environments. Although both technologies have seen individual advancements, their integration remains limited. Most hexapods still rely on manual control or simple pre-programmed actions. This project aims to bridge that gap by developing a hexapod robot that responds to real-time voice commands using AI, enhancing both functionality and user experience in practical applications.

II. Objectives

The primary objective of this project is to design and develop a hexapod robot capable of stable and efficient multi-legged locomotion. The system aims to integrate AI-based speech recognition to enable real-time voice control, allowing intuitive and hands-free interaction. Inverse kinematics will be implemented to achieve smooth and coordinated leg movements. The robot will feature wireless communication for remote operation and monitoring. A key goal is to enhance human-robot interaction by supporting natural language commands. Additionally, the system will be tested under various terrain and environmental noise conditions to evaluate its performance. The project also emphasizes creating a modular and scalable architecture to support future enhancements and applications.

III. Research and Methodology

3.1 Hardware Design

3.1.1 Chassis and Leg Structure

The hexapod robot features a six-legged configuration, designed for stability and efficient movement across uneven terrains. Each leg consists of multiple joints—typically three per leg (hip, knee, and ankle)—which provide flexibility and allow for complex movements. The legs are made from lightweight, durable materials, such as high-strength plastic or aluminium, to ensure both strength and minimal weight for better performance.

3.1.2. Actuators (Servo Motors)

Each of the six legs is powered by servo motors, which are responsible for precise movement and control of the leg joints. The number of motors required depends on the number of joints in each leg, typically three motors per leg (one for each joint). These motors provide the necessary torque and precision to achieve coordinated locomotion.

3.1.3. Sensors

The robot is equipped with various sensors to maintain balance and facilitate autonomous navigation:

□ Accelerometers and Gyroscopes: These sensors help detect the robot&#39;s orientation and acceleration, providing feedback for balance and stability, especially on uneven terrain.

□ Force Sensors: Used to detect contact with the ground and ensure that the robot is stable while walking.

□ Distance Sensors (Ultrasonic or Lidar): These help with obstacle detection, enabling the robot to avoid collisions and navigate effectively.

3.1.4. Microcontroller and Processing Unit

A microcontroller, such as an Arduino, Raspberry Pi, or a more advanced embedded system, serves as the central processing unit (CPU) for the robot. It processes the voice commands, runs the inverse kinematics algorithm to control leg movements, and interfaces with the sensors and actuators. The microcontroller also handles wireless communication with the remote interface.

3.1.5. Wireless Communication Module

The robot is equipped with wireless communication capabilities (e.g., Wi- Fi or Bluetooth) to enable remote control and monitoring. This allowsthe operator to send voice commands and receive feedback, making the system more flexible and interactive.

3.1.6. Power Supply

The hexapod robot requires a battery to power the motors, sensors, microcontroller, and communication modules. Typically, rechargeable lithium-ion or lithium-polymer batteries are used for their high energy density and long lifespan. The power system is designed to provide enough voltage and current to operate the motors and other components simultaneously.

3.1.7. Speech Recognition Hardware

For the voice control system, the robot is equipped with a microphone or a microphone array to capture user commands. The audio signals are then processed using a speech recognition module, which can be implemented using a low-power embedded speech recognition system (e.g., Raspberry Pi with a USB microphone or a specialized speech recognition module like Vosk or PocketSphinx).

3.2 Software Architecture

3.2.1. Speech Recognition System

The core software component for voice control is a speech recognition module, which converts spoken commands into text that the robot can interpret. This is typically achieved using deep learning models trained for Natural Language Processing (NLP). The software used can be either a pre-trained model or an offline recognition tool such as PocketSphinx or Vosk, which work well for embedded systems with limited processing power. The microphone captures the user’s voice, and the software processes the audio to identify commands such as "move forward," "turn left," or "stop."

3.2.2. Natural Language Processing (NLP)

Once the speech recognition system converts audio into text, NLP algorithms are applied to parse the commands and determine their meaning. The software must handle different accents, noisy environments, and variations in phrasing. This component uses keyword extraction and command interpretation to ensure the robot can understand a range of voice commands, including simple navigation commands (e.g., "go forward") or more complex instructions (e.g., "turn 45 degrees left").

3.2.3. Movement Control and Inverse Kinematics

The software uses inverse kinematics (IK) algorithms to translate high-level movement commands into low-level leg movements. For example, when the user says, "move forward," the software calculates the precise angles and positions required for each of the six legs to achieve that movement. This involves solving the kinematic equations for each leg to ensure smooth, coordinated movement. The software then sends control signals to the servo motors, adjusting the position of each leg accordingly.

The IK calculations ensure that the hexapod moves efficiently and maintains balance, even when navigating uneven terrain. The software continuously updates the leg positions based on real-time feedback from the sensors.

3.2.4. Sensor Integration and Feedback Processing

The robot is equipped with accelerometers, gyroscopes, and distance sensors, which provide feedback to the software. This feedback is essential for balancing and avoiding obstacles. The sensor data is processed by the software to monitor the robot's orientation and adjust the movements if needed (e.g., if the robot begins to tip over or encounters an obstacle).

3.2.5. Wireless Communication and Remote Control

The software supports wireless communication between the robot and a remote interface (e.g., mobile app or desktop). This communication allows the robot to send status updates and receive commands. The remote interface acts as the control center, where the user inputs voice commands or manually adjusts settings. This system is built using standard communication protocols like Wi-Fi or Bluetooth, enabling two-way communication between the robot and the user interface.

3.2.6. Control Loop and Real-Time Processing

The software must run in a continuous real-time control loop to ensure smooth operation. Each loop processes inputs from the microphone, interprets them using the NLP system, performs inverse kinematics calculations, and sends commands to the actuators. It also handles sensor feedback, adjusting leg movements to maintain balance and stability.

The control loop runs on the embedded microcontroller or Raspberry Pi (or similar hardware), where the software is optimized to minimize latency and ensure fast response times. The robot can process multiple voice commands and update its movement almost instantaneously.

3.2.7. User Interface

For interaction, the software includes a mobile app or desktop interface for monitoring and controlling the robot remotely. This interface allows the user to configure settings, visualize sensor data, and issue commands. The user interface is typically built using frameworks like Flutter for mobile apps or Python (Tkinter or PyQt) for desktop applications.

3.2.8. Error Handling and Optimization

The software includes error-handling mechanisms to manage unexpected conditions, such as incorrect voice input, sensor malfunctions, or loss of wireless communication. It also features optimization algorithms to enhance performance in various environments, such as adjusting the robot’s walking pattern for smoother movement over rough terrain.

3.3 Implementation

3.3.1 Mechanical Assembly

The hexapod robot was built using a lightweight frame made from acrylic or aluminum. Each of the six legs was designed with three degrees of freedom (DOF), requiring a total of 18 servo motors. These were mounted at the hip, knee, and ankle joints to enable complex walking and turning movements.

3.3.2. Servo Motor Control

An Arduino board was used to control the 18 servo motors. It received movement instructions from the Raspberry Pi and generated precise PWM signals to operate each joint. This separation of processing (Raspberry Pi) and actuation (Arduino) ensured smooth and synchronized motion.

3.3.3 Processing Unit

A Raspberry Pi acted as the brain of the robot. It handled voice recognition, command parsing, and coordination of leg movements. The Pi also interfaced with the Arduino, sensors, and microphone, making it central to all data processing and control.

3.3.4 Speech Recognition

Voice commands were captured using a USB microphone connected to the Raspberry Pi. The Vosk speech recognition engine, running offline, was used to process voice input and convert it into text. This ensured real-time command recognition without the need for internet access.

3.3.5 Natural Language Command Parsing

The recognized text was processed using a Python script to extract relevant command keywords. Based on the identified command—such as "forward," "left," "stop"—corresponding actions were triggered in the robot's motion control module.

3.3.6 Inverse Kinematics for Motion

An inverse kinematics (IK) algorithm was implemented to compute the required joint angles for each leg based on the desired direction and speed of movement. This enabled smooth walking and turning by synchronizing leg movements precisely.

3.3.7 Sensor Feedback and Balance Control

The robot used an IMU sensor (gyroscope and accelerometer) to monitor orientation and movement. This sensor data was processed to make real-time adjustments to leg positions, ensuring balance, especially on uneven surfaces.

3.3.8 Wireless Communication

Wi-Fi and Bluetooth modules were added to allow communication between the robot and a user interface. This enabled remote monitoring, manual control, and debugging using a mobile app or PC- based dashboard.

3.3.9 Integration and Testing

After assembling all hardware and implementing software modules, the entire system was integrated. The robot was tested in various environments to check voice recognition accuracy, response time, walking stability, and obstacle handling.

3.4 Working

The AI voice-controlled hexapod robot operates through a multi-stage process combining voice recognition, signal processing, and robotic locomotion. The system begins with the user giving a voice command through a connected microphone. This audio input is sent to a Raspberry Pi, which uses the Vosk offline speech recognition engine to convert speech to text. The recognized text is parsed by a Python script to detect key commands like "forward," "backward," "left," or "stop."

Once a command is identified, it is translated into a motion instruction and sent to an Arduino microcontroller over a serial connection. The Arduino is responsible for controlling 18 servo motors—3 on each leg—through a servo driver module. Inverse kinematics calculations determine the angles required at each joint to produce coordinated movement. The Arduino generates PWM signals to actuate the servos accordingly.

The robot walks using a tripod gait pattern, where three legs move at a time while the other three provide support, ensuring stability. Real-time feedback is obtained from IMU sensors (gyroscope and accelerometer), which help maintain balance and correct posture. If the robot tilts or slips, it automatically adjusts the leg positions for recovery. Wireless communication via Wi-Fi or Bluetooth allows users to monitor robot status or override commands using a mobile app or computer. The entire system works in a loop: receiving voice input, processing the command, executing movement, and adjusting based on feedback. This enables the robot to respond intelligently to voice commands and navigate complex environments with stability

IV. Conclusion

The development of the AI voice- controlled hexapod robot demonstrates the successful integration of speech recognition, robotic locomotion, and intelligent control systems. By combining mechanical design with modern AI technologies, the robot can interpret voice commands and perform complex walking patterns with stability and precision. The use of inverse kinematics and sensor feedback ensures smooth and adaptive movement, even on uneven surfaces.

This project showcases how voice- controlled systems can enhance human-robot interaction, making robotics more intuitive and accessible. The offline speech recognition system ensures reliability without dependency on internet connectivity, which is crucial for field applications. Wireless communication, modular architecture, and real-time control features further contribute to the robot's flexibility and usability in different environments.

Looking forward, the robot presents numerous opportunities for advancement in both hardware and software. With future upgrades like vision systems, multilingual support, and autonomous navigation, this platform can be extended for use in fields such as disaster response, surveillance, space exploration, and assistive robotics. The project stands as a strong foundation for future innovations in intelligent legged robotic systems

The AI voice-controlled hexapod robot has significant potential for enhancement and broader applications. Future versions can integrate computer vision using cameras and AI models to enable autonomous obstacle detection and path planning. Multilingual voice recognition can be added to support a wider range of users. Improved Natural Language Processing (NLP) will allow the robot to understand more complex and conversational commands. Integration with IoT and smart home systems could enable it to assist in domestic automation tasks.

Adding GPS modules would allow outdoor navigation and location tracking. Solar-powered charging systems could enhance the robot's energy efficiency and runtime in field conditions. Gesture and facial recognition could provide more intuitive human-robot interaction.

V. References

1. Sharma, A., &amp; Sharma, V. (2020). Design and Control of Hexapod Robots – A Review. International Journal of Robotics Research, 8(2), 45– 52.
2. Jadhav, A. S., &amp; Kale, R. R. (2019). Voice Controlled Robot using Android Application. International Research Journal of Engineering and Technology (IRJET), 6(5), 1257–1260.
3. Ramesh, M., &amp; Srinivas, G. (2021). Arduino Based Voice Controlled Wheeled Robot using Bluetooth Module. International Journal of Engineering Research &amp; Technology (IJERT), 10(7), 85–88.
4. Lee, J., Kim, D., &amp; Park, S. (2018). Gait Planning and Control for Hexapod Walking Robots on Uneven Terrain. Robotics and Autonomous Systems, 104, 56–65.
5. Mishra, R., &amp; Singh, N. (2020). Servo Motor Based Robotic Arm Controlled via Voice Recognition. International Journal of Engineering and Advanced Technology (IJEAT), 9(6), 78–83.
6. Sharma, V., &amp; Arora, S. (2019). Smartphone Controlled Robot Using Arduino and Bluetooth. International Journal of Scientific Research and Engineering Development, 2(3), 437–441.

7. Patel, A., &amp; Chavda, J. (2022). Development of Arduino Based Hexapod Robot with Real-Time Gait Execution. Journal of Mechatronics and Automation, 11(1), 33–38.

8. Khan, M. A., &amp; Sheikh, Z. A. (2021). Artificial Intelligence Based Speech Recognition for Robotic Applications. International Journal of Computer Applications (IJCA), 174(3), 1–5.

9. Prasad, D. &amp; Kumar, A. (2020). Implementation of Real-Time Video Surveillance Using ESP32-CAM and IoT Cloud. International Journal of Scientific &amp; Engineering Research (IJSER), 11(10), 1534–1538

10. Prasad, D. &amp; Kumar, A. (2020). Implementation of Real-Time Video Surveillance Using ESP32-CAM and IoT Cloud. International Journal of Scientific &amp; Engineering Research (IJSER), 11(10), 1534–1538.