Bondhu: An IoT-Enabled Humanoid Robot with Real-Time Voice Control and Face Recognition Capabilities

-Aka Das[*1]
-Anandip Barua[*2]
-Md. Hasanuzzaman[*3]

Abstract— The convergence of the Internet of Things (IoT), computer vision, and natural language processing (NLP) has hastened the development of intelligent robotic systems capable of interacting with humans and the environment seamlessly. This paper presents a novel IoT-based robotic platform incorporating real-time voice control and adaptive face recognition to improve autonomy, contextual awareness, and user interaction. The system architecture includes an Arduino UNO microcontroller, servo actuators, microphone, speaker, LCD display, camera module, and onboard processing unit for local processing. The robot can recognize voice commands and change its behavior dynamically based on the recognized person using speech recognition and deep learning-based face recognition, which makes the operation hands-free and intuitive. Onboard speech emotion recognition was achieved using Edge Impulse, and cloud-based environmental monitoring and data analysis were realized using ThingSpeak and Microsoft Power BI. Experimental trials validate high command recognition accuracy, strong face detection under varying light conditions, and successful completion of high-level tasks such as object detection, navigation, and human interaction. This architecture is modular, low-cost, and extensible, with excellent potential value for application in assistive robotics, home automation, and factory automation. This research brings human-robot interaction to the next level by showing how IoT and AI can be integrated in a comprehensive manner to create smart, interactive robots.

Keywords— Humanoid Robot, Internet of Things, Computer Vision, Natural Language Processing, Large Language Model

[*1]Lecturer, Department of CSE, Cox's Bazar International University, Bangladesh

[*2]Lecturer, Department of CSE, Cox's Bazar International University, Bangladesh

[*3]Lecturer, Department of CSE, Cox's Bazar International University, Bangladesh

I. INTRODUCTION

The integration of robotics, the Internet of Things (IoT), computer vision, and natural language processing (NLP) continues to transform human-machine interaction paradigms. This interdisciplinary convergence enables a new generation of cognitive robotic systems to interpret and respond to voice commands, facial recognition, and emotional cues with higher contextual awareness and accuracy. Such systems are evolving from traditional mechanistic behavior to exhibit cognitive and adaptive features and to support more intuitive and natural interactions with users. According to the 2023 World Robotics Report, global installations of service robots increased by 37% from 2021 to 2023, of which vision- and voice-equipped smart robots comprised the fastest-increasing category [1],[2]. This trend indicates a growing global demand for multimodal robotic systems that will facilitate advanced, transparent human-robot interaction.

These technologies have enormous impacts in a broad spectrum of applications ranging from health care assistance to home automation, security, and industrial process optimization. According to the IEEE Robotics and Automation Society, face- and voice-controlled robots exhibit considerably greater task achievement efficiency compared to conventional systems [3]. Further, assistive robotics has been identified as a field in which voice-interactive systems are capable of restoring autonomy to mobility-impaired individuals. Recent research by the International Federation of Robotics has revealed that voice-operated assistive robots could help people with severe physical disabilities achieve independent living to a considerable degree, thereby fulfilling a growing societal need [4].

The integration of IoT infrastructure with robot platforms provides these aspects with an additional push through distributed intelligence, cloud-based control, and remote monitoring. Today, that is, as of early 2025, the number of IoT devices connected in the world stands at around 43 billion with robot systems comprising slightly over 8% of the overall ecosystem, as per Gartner [5]. Such widespread connectivity enables real-time data collection, distributed decision-making, and responsiveness to the environment—indicators of successful implementation in dynamic operational environments [6]. According to the 2024 McKinsey Global Institute report, IoT-enabled robotic systems can add between $1.9 trillion and $3.7 trillion of economic value each year by 2030, particularly through enhanced efficiency, autonomous services, and reduced operational costs.

Advancements in communication technologies—most notably the proliferation of 5G networks—are enabling the low latency, high-speed data transmission required for real-time robotic coordination and cloud-IoT integration. These technology facilitators are critical to application domains such as precision agriculture, automated logistics, and smart city infrastructure, where responsiveness and scalability are key [7],[8]. In this rapidly evolving field, the present work introduces a novel IoT-enabled robotic system that synergistically integrates real-time voice command control, facial recognition via deep learning, and environment-conscious intelligence.

The system demonstrates how low-cost, off-the-shelf hardware building blocks like Arduino UNO microcontrollers, MEMS microphone modules, servo motors, and wireless camera modules can be correctly interfaced with smart software to provide autonomous, adaptive, and user-aware robotic platforms. Recent studies at MIT's Media Lab indicate that systems with the capacity to understand user context and emotional state can reduce cognitive load on human operators by up to 64%, increasing accessibility for non-experts and individuals with varying levels of technological literacy [9],[10]. Despite these advances, there are important challenges in developing robotic systems that operate reliably under real-world conditions of varying illumination, ambient noise, communication latency, and inconsistent recognition of emotional expression.

This study addresses these challenges by designing an autonomous robotic platform with real-time multimodal interaction. Specifically, it decodes voice commands, recognizes emotional vocal expressions, and recognizes human faces using deep learning algorithms tailored for embedded systems. The research focuses on three primary goals: one, the design and deployment of a facial recognition system using lightweight deep learning models for real-time inference on edge devices; two, the development of a robust, on-device speech emotion recognition pipeline that is trained using Edge Impulse for reliable operation under different acoustic environments; and three, the implementation of IoT protocols and services like ThingSpeak and Microsoft Power BI for real-time data sharing, remote system monitoring, and performance analysis.

The proposed robot system relies on modularity and adaptability that can be applied in various fields of application, including assistive technology for the disabled and older people, intelligent home systems, and collaborative industrial environments. The paper delivers a set of contributions to the field of embedded artificial intelligence and human-robot interaction. First, it provides a homogenous multimodal interaction platform consisting of voice command processing, emotion speech analysis, face recognition, and IoT networking for enabling context-aware, naturalistic interaction. Second, edge deployment of emotion recognition models to resource-constrained microcontrollers on Edge Impulse minimizes latency and mitigates reliance on public cloud resources, increasing system response and data security. Third, their integration in thermal sensors, real-time streaming video, and cloud-analytic platforms supports intelligent environmental monitoring and reactive robotic action. Fourth, hardware design uses readily available parts, maintaining extremely low implementation costs and facilitating scale-up and customization. Fifth, the system demonstrates the potential for personalized interaction according to user identity and affective state, enhancing engagement and operational efficacy.

Compared to prior embedded robotic platforms, *Bondhu* advances the state of the art by combining multimodal interaction (voice, vision, and emotion), on-device deep learning, and IoT-based environmental analytics into a unified and scalable system optimized for real-world, resource-constrained deployment. This research is organized into several significant sections. The Introduction provides the context and technological significance of the study and establishes its scope and objectives. The Literature Review critically reviews the current research in multimodal robotic control systems, voice and face recognition, and robotics in the context of IoT, identifying the gaps addressed in this work. The Methodology chapter describes the system architecture, hardware configuration, software frameworks, and integration strategies adopted in the development of the robotic platform. The Results and Discussion chapter presents empirical tests of system performance, e.g., recognition accuracy, response time, environment flexibility, and power savings, and analysis of challenges encountered and mitigation techniques. Finally, the Conclusion capture the contributions of the work, offer limitations, and propose potential future enhancements, e.g., in terms of incorporation of more advanced machine-learning and extension of IoT functionality to be employed more broadly.

II. Related Work

There is a vast body of literature that addresses voice-controlled robotic systems and their implications for natural human-robot interaction. Jnr, B. A. [11] created a voice-controlled robotic car that demonstrates the appropriateness of voice inputs in offering effective navigation and task completion. This study demonstrates how speech interfaces can make robot control easier, especially for non-technical individuals. Holubek et al. [12] focused on verifying voice control modifications for the DOBOT Magician robot, specifically examining how voice frequency changes affect control accuracy and system performance. This work

contributes to understanding the technical parameters necessary for reliable voice-controlled robotic operations. Building on voice control foundations, Piyaneeranart and Ketcham [13] developed an automatically moving robot intended specifically for elderly users, incorporating voice control mechanisms to enhance mobility assistance. Similarly, Gupta [14] proposed a novel voice-controlled robotic vehicle designed for smart city applications, expanding the scope of voice-controlled robotics beyond individual assistance to urban infrastructure integration.

Along with the progress of voice recognition, facial recognition has become a standard feature in robot systems for identity verification, adaptive interaction, and security. Kamilaris and Botteghi [15] argue that the integration of IoT with robotics and biometric recognition can lead to highly customized service robots that learn to identify individual users through facial recognition. Safety and learning mechanisms are essential for autonomous robotic systems. Kiangala and Wang [16] implemented a safety response mechanism for autonomous moving robots in small manufacturing environments, utilizing Q-learning algorithms combined with speech recognition to enable adaptive safety responses based on voice commands and environmental conditions.

Supporting these intelligent robotic systems is an ecosystem of sophisticated sensors and decision algorithms. Nanade and Anne [17] conducted a comparative analysis of hybrid IoT autonomous robotics, specifically comparing LiDAR-based precision mapping with camera-based vision systems using ROS2 and MediaPipe platforms. This research demonstrates how different sensing modalities can be integrated to enhance robotic perception and autonomous navigation capabilities. Advanced command processing represents another crucial aspect of voice-controlled systems. He et al. [18] developed an attention-based command detection model that enables natural language processing in voice control systems, allowing robots to interpret more complex and conversational commands rather than simple predetermined phrases. This advancement significantly improves the user experience by making human-robot interaction more intuitive and natural.

The application of machine learning for progressive improvement has been explored in various contexts. TV and Udupa [19] developed a voice-controlled 6 degrees of freedom (DoF) arm mobile robot specifically designed for assisted home environments, demonstrating how voice control can be applied to complex manipulator systems for domestic assistance tasks.

No-code programming approaches have emerged as a significant trend in robotic development. Halim et al. [20] introduced a markerless approach for multimodal natural interaction in human-robot collaboration contexts, enabling agile production scenarios through no-code robotic programming. This approach democratizes robot programming by allowing non-technical users to configure robotic systems through intuitive interfaces.

In medical applications, Rogowski [21] developed scenario-based programming methodologies for voice-controlled medical robotic systems, addressing the specific requirements of healthcare environments where precise control and safety protocols are paramount. This work demonstrates how voice control can be safely integrated into sensitive medical applications while maintaining the reliability and accuracy required for healthcare robotics.

Security and user privacy are central concerns in robotic systems utilizing biometric data. This work briefly explores these challenges in the context of IOT-connected, face-recognizing robots. Yang et al. [22] provide an extensive review of biometric technology in IoT-based security systems, with a discussion of how facial recognition can be used as an effective authentication tool in various robotic applications. Taking this concept further, Beyrouthy et al. [23] researched EEG-based biometrics as a further layer of security and

customization, validating the need for multimodal biometric approaches in enhancing user experience and system resilience.

The intersection of IoT and robotics has created the Internet of Robotic Things (IoRT), a theoretical and practical framework outlined by Krejrí et al. [24] The IoRT paradigm views robots as intelligent nodes in a distributed system, capable of autonomous operation and data sharing. The IoRT model is especially suited to facial recognition systems, which require real-time data processing and cross-device synchronization. Pradhan et al. [25] also highlight the trend in the healthcare industry where IoT-based robotic systems have been used for surgery assistance and rehabilitation via voice interaction and face recognition technology.

Face recognition in robots has also been used for system access and control of the environment in non-healthcare applications. Firdayanti et al. [26] also proposed an IoT-based electronic device management system with face recognition to enhance operational security and surveillance. These are examples of bigger trends in integrating facial biometrics into everyday robotic use.

Literature shows a powerful trend towards networked, multimodal robots based on IoT connectivity, voice interfaces, and facial recognition to deliver flexible, secure, and user-controlled experiences. Yet even with these developments, there remain significant integration gaps—most work to date examines these pieces in isolation or within tightly bounded domains such as healthcare, smart cities, or industrial automation. Issues of environmental variance, edge processing limits, and biometric recognition variability still need to be fully solved. *Bondhu* addresses such gaps by proposing one, inexpensive robot system that brings voice recognition, facial recognition, and control of IoT devices all into a general-purpose, offline-compatible framework. It is uniquely optimized for personalized interaction in educational or institutional environments, possessing a context-sensitive knowledge base that supports customized replies. In addition, *Bondhu* is underscored as being convenient and adaptable, offering near-real-time performance on modest hardware while being stable under varying environmental conditions. In filling the gap between experimental research platforms and real-world deployments for non-experts, *Bondhu* incorporates a modular, scalable design that encourages reproducibility, tunability, and future expansion in intelligent robotics.

III. METHODOLOGY

This section discusses the structured development and unification of the face recognition-capable and voice control-capable IoT-based robot system. The method is a modular design approach having hardware assembly, software implementation, communication interface setup, and real-time human-robot interaction algorithmic implementation. The entire method ensures that all the subsystems—voice recognition, face detection, motor control, and IoT connectivity—are running in absolute synchronization in the robotic arrangement.

A. Dataset Collection and Management

The *Bondhu* robot implements a sophisticated dual-dataset approach that underpins both its facial recognition capabilities and conversational intelligence. The dual-dataset approach refers to the use of two distinct data systems: one for storing facial embeddings used in identity recognition, and another for managing conversational logic through both predefined responses and generative AI. *Bondhu* implements this by maintaining a serialized database of facial vectors for recognition tasks and combining rule-based queries with PaLM-generated replies to handle diverse verbal interactions efficiently. These datasets are structured to balance performance, storage efficiency, and real-time responsiveness in a university environment.

The facial recognition system employs a vector-embedding approach rather than traditional image storage. When new users are enrolled through the voice command "add new user," the

system initiates a multi-step process that prioritizes data efficiency and recognition accuracy. The webcam captures a sequence of facial frames (typically 20–30 images) from slightly different angles and expressions to ensure robust recognition. For each frame, the system applies the face_recognition library's deep learning model to extract a dense 128-dimensional facial embedding vector. The 128-dimensional facial embedding vector is a compact numerical representation that captures the unique geometric and texture-based features of a person's face. It is generated by applying a pre-trained deep learning model to each captured frame, which encodes the facial characteristics into a fixed-length vector suitable for fast and accurate identity matching. These embeddings represent the distinctive geometric and texture features of the individual's face, mathematically encoded for efficient comparison.

Each embedding vector is paired with identifying metadata including the person's name, position (student/faculty/staff), and department affiliation. This composite data structure is serialized and appended to the existing face_data.pkl file using Python's pickle module, creating a persistent and expandable database. The system maintains referential integrity by implementing a unique identifier for each individual, allowing for future updates to biographical information without disrupting the recognition capabilities. The decision to store embeddings rather than raw images provides multiple benefits: significantly reduced storage requirements (approximately 4KB per person versus several MB for image sets), faster comparison operations during recognition, and enhanced privacy as the original facial images aren't retained in permanent storage. During recognition, the system computes the L2 (Euclidean) distance between a newly captured face embedding and all stored embeddings, applying a threshold-based classification approach with a default confidence threshold of 0.6 to balance false positives and negatives.

The conversational intelligence of *Bondhu* operates on a hybrid dataset model with two distinct components that work in tandem to provide relevant responses:

The system maintains a structured, rule-based knowledge base implemented directly in the code through conditional statements. This knowledge base contains approximately 50-60 predefined query patterns mapped to specific institutional information about Cox's Bazar International University. Each entry follows a pattern-matching approach where variations of similar questions ("who is the chairman", "current chairman", etc.) map to the same response through logical OR conditions in the code. This knowledge base prioritizes high-accuracy responses for domain-specific information, including university history (founding in 2013), organizational structure (identifying the chairman, secretary, and faculty members), departmental information, and student cohort details. Pattern matching uses case-insensitive substring matching rather than exact matching, providing flexibility in query formulation.

For non-matching queries, *Bondhu* takes a generative AI approach by using Google's PaLM API. The system sends non-matching queries to llm_model_face function, which builds a prompt with limiting stipulations ("Give a short response") and temperature settings (0.3) to get contextually appropriate answers. The temperature setting of 0.3 is a deliberate design to limit randomness in the language model's output to generate consistent and fact-grounded answers. This controlled generation methodology is within the scope of the education environment of the system, where correctness and reliability precede creativity. In doing so, the robot can handle a theoretically unlimited number of questions beyond its immediate knowledge base. Temperature setting of 0.3 is a deliberate design choice to favor more deterministic, factual responses over creative but less accurate ones, more appropriate in the environment of an educational institution where information accuracy is of utmost value. The two data sets complement each other for a smooth interaction experience, with the system attempting to match against learned question patterns first before falling back on the generative AI approach. This design finds a middle ground between the precision of pre-

programmed messages and the flexibility of AI-based content, with facial recognition offering customized interaction based on the identity of the user interacting with *Bondhu*.

B. System Architecture and Functional Flow

The system architecture is a combination of processing units, actuators, sensors, and embedded microcontrollers. The top-level system is centered on the Arduino UNO microcontroller, which serves as the actuator driver and the command interpreter. Voice commands given by a user are captured by a microphone and interpreted using an associated smartphone application or cloud-based speech-to-text API. Processed commands are transmitted to the Arduino via Bluetooth or Wi-Fi.

A miniature computer such as Raspberry Pi is used as the primary computer unit to execute more computationally intensive operations such as real-time face recognition. A camera module is used to capture real-time video streams, which are processed using methods of computer vision. Depending on the recognized command or face, the Arduino directs the corresponding motors to execute corresponding operations, such as movement or gesture. A speaker and LCD display are integrated to provide audio and visual feedback to consumers. Modular communication is facilitated using standard protocols such as I²C, UART, and Bluetooth to allow low latency as well as reliable data transfer. In this project, the LCD display communicates with the Arduino using the I²C protocol for efficient real-time visual feedback, while UART is used for serial communication between the Arduino and the Raspberry Pi to transmit command data. Bluetooth connectivity enables wireless communication between the robot and a smartphone or external control device for remote voice command input. Figure 1 shows our system architecture and Figure 2 shows our flow diagram of the work.

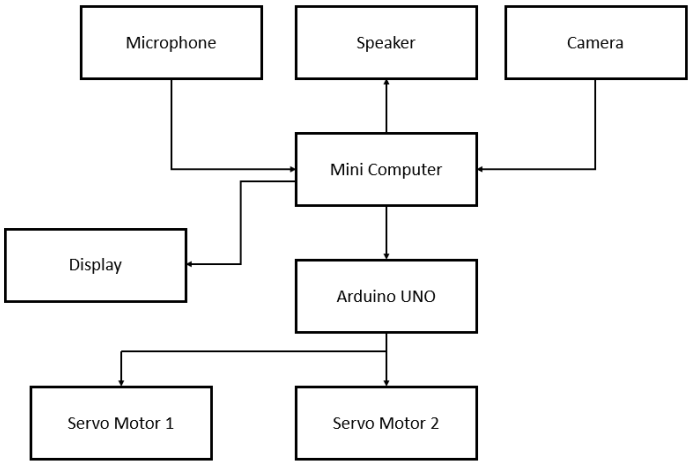

Fig. 1 Proposed System Architecture

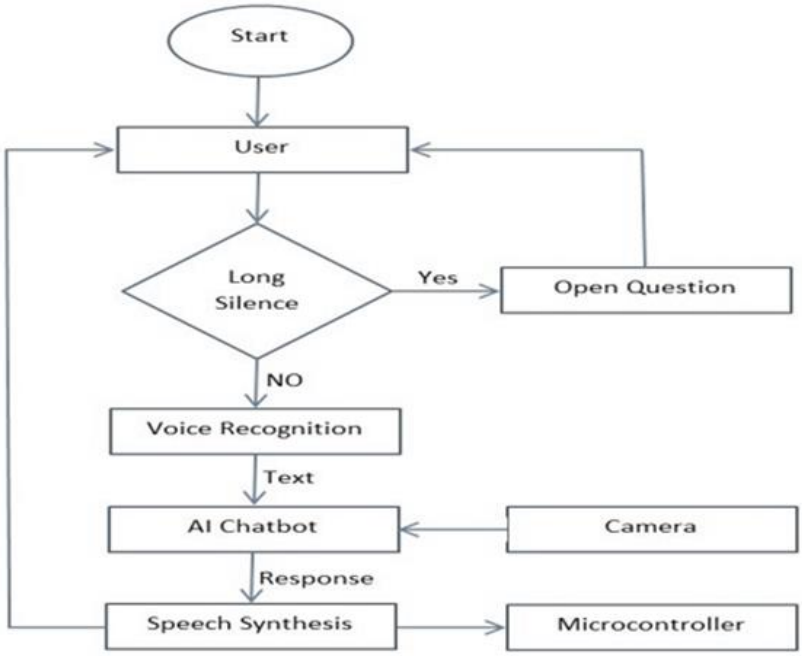

Fig. 2 Proposed System Flow Diagram

The software architecture of the robot is developed using Python 3.10, chosen for its rich ecosystem of libraries in computer vision, speech processing, and AI integration. The project uses OpenCV for camera interfacing and frame manipulation, face_recognition (built on dlib) for extracting and matching facial encodings, and speech_recognition for real-time audio command processing. Voice feedback is handled using pyttsx3 for offline synthesis and optionally gTTS for high-quality cloud-based TTS. Integration with Google's PaLM API enables generative responses to open-ended questions, handled through the google.generativeai module. In the flow diagram (Figure 2), the block labeled "AI Chatbot" corresponds to the generative AI module powered by Google's PaLM API, which handles open-ended or unmatched queries by generating contextually appropriate responses, effectively serving as the system's conversational engine. The application follows a modular structure where all hardware-triggering functions (such as motor movements) are encapsulated in helper modules, and the main control loop handles interpretation, execution, and fallback logic. This software stack ensures adaptability, expandability, and a clean separation between hardware control and high-level reasoning.

C. Hardware Implementation

All the parts are mounted on a rigid chassis to form the physical body of the robot. Servo motors are mounted on joints and facial features to emulate simple humanoid gestures. The camera is mounted at head level to emulate eye-level vision, which enhances the accuracy of face detection. The 1080P Wi-Fi IR night vision camera is strategically mounted at the head level of the robot, emulating human eye positioning to optimize facial detection and recognition accuracy by maintaining a natural line of sight with interacting users. Table I provides an overview of the key hardware parts used in the system and their roles and Figure 3 shows the component pictures.

TABLE I
Hardware Components and Functional Roles

Component	**Description**	**Function**
Arduino UNO	ATmega328P-based microcontroller	Controls actuators and manages I/O signals
DS3218 Servo	20 kg·cm torque digital servo	Handles heavy-duty robotic

Motor	with metal gears	movements
Mini Computer	Raspberry Pi	Runs face recognition and communication stack
BOYA M1DM Microphone	Omnidirectional lavalier microphone	Captures user voice input
1080P Wi-Fi Camera	High-resolution IR night vision capable camera	Captures live video for face recognition
Speaker	Standard speaker compatible with Arduino	Provides voice output and feedback
LCD Display	SPI/I²C-compatible SSD1306 OLED display (128×64 pixels)	Displays system status and face recognition
Power Supply	External regulated power (7–12V input)	Powers the system components

Fig. 3 System Used Component Pictures

D. Arduino UNO Control Hub

The Arduino UNO is selected since it is highly flexible, provides real-time processing, and is compatible with a vast array of sensors and modules. It operates on 5V and provides 14 digital I/O pins out of which six can provide PWM as well as six analog inputs. Communication is enabled using in-board UART, SPI, and I²C interfaces. This provides end-to-end communication with external modules such as motor drivers, LCDs, and Bluetooth transceivers. The Arduino responds to commands and performs actions on them by generating PWM signals to control servo angles, as guided by facial recognition or voice output.

E. Control of Servo Motor

The robot uses two servo motors for performing the physical movements: a plastic gear servo constructed of lightweight plastic for head turns or facial movements, and a DS3218 digital metal gear servo for functions that require more power, such as handshakes or arm lifting. The servos are controlled by PWM signals from the Arduino that determine the angle of rotation between 0° to 180°.

For servo positions to be updated in real-time for giving natural and responsive motion, user input in the form of recognized voice command or detected faces is employed. For

example, while a user says "handshake" or a known face is seen, an immediate signal goes out to rotate the servo to the corresponding gesture position. All movement sequences have been designed to be executed within a window time so that smoothness is assured and motor overheating is prevented.

Reliability and safety are achieved through simple constraints in the control code. Servos are commanded not to exceed their mechanical limits, and brief stops in between repeated motions are given to reduce wear. Power consumption is also brought under control, and servos are powered back to a resting state after each action to conserve power. These measures make the actuator system not only functional but sustainable for repeated applications within interactive settings.

F. Mini Computer for High-Level Processing

A computer-on-board unit like the Raspberry Pi is employed as the computing platform for computer-intensive tasks. It is employed in image acquisition, face detection, and network communication. The boards are sufficiently powered to run Python-supported OpenCV or TensorFlow Lite models with GPIO support for Arduino communication. The mini-computer is also utilized to decode Wi-Fi-based camera streams, pre-process images, and conduct face detection either locally or using a cloud API.

G. Voice Input and Output System

The BOYA M1DM dual-input lavalier microphone is used for obtaining voice commands. It provides high-fidelity omnidirectional audio input and is designed for hands-free real-time communication. Obtained audio is either transmitted to the smartphone software or processed directly by the mini-computer for converting to text. The speaker system finishes the job by delivering audio feedback, such as the confirmation of recognized commands or a greeting when a face is recognized. Integration with text-to-speech (TTS) engines supports natural-sounding output.

H. Visual Feedback through LCD Display

An SPI/I²C-compatible LCD screen with a resolution of 128×64 pixels (such as the SSD1306 OLED display) is integrated into the system to provide real-time visual feedback, including recognized face IDs, system status, and command prompts. This display is fully supported by the Arduino UNO, which receives formatted serial messages (e.g., Display:<text>) from the mini-computer, parses them, and updates the screen accordingly. The SSD1306 uses the I²C protocol and requires minimal memory, making it well-suited for the UNO's 2 KB SRAM. This offloads basic visual tasks to the Arduino while reserving the mini-computer for computationally intensive operations like face recognition and voice processing. If implemented, the display’s backlight or contrast can be adjusted through the Arduino firmware based on ambient input or system state.

I. Face Recognition Camera System

The system has facial recognition and environmental monitoring functions that are supported by a 1080P Wi-Fi camera with an infrared night vision, a 160° wide-angle lens. It can encode video using H.264, and can record video onto a microSD card. Wireless transmission of video frames to the mini-computer where face detection and recognition algorithms are executed is done via a local network. Real-time facial detection and recognition is done using open-source libraries as OpenCV, Dlib, and other deep learning frameworks. The faces that are detected are associated with specific actions or responses that have been pre-programmed, and are acted upon.

J. Communication Protocols

The system uses various standard communication interfaces to integrate its hardware components. The initial connection of the mini-computer which is a Raspberry Pi 4 Model B and the Arduino UNO is through a serial UART communication link using a 9600 baud rate. Specific command strings such as “1”, “2”, “3”, and “4” are used to invoke the corresponding movements of the mouth and handshakes. Peripheral modules like the LCD screen have previously defined I²C or SPI interfaces, which provide low latency peripheral visual feedback. Bluetooth and Wi-Fi modules are optional peripherals and can be used for remote control via smartphone. The system uses a protocol which ensures command and control integrity, prevents signal interference, and optionally implements acknowledgment replies to improve fault tolerance.

K. Voice Recognition and Natural Language Processing Pipeline

The voice interaction pipeline is designed to understand spoken input in a timely manner and in a contextually relevant way. It works with a two layers logic: command-level response and a backup dialogue system with conversational flow. Speech recognition evaluates and attempts to match utterances with a set of command keywords and phrases mapped to specific robot actions, including face recognition and reporting time/date, as well as some motor gestures. The efficiency of this form of speech recognition is that when a user frequently issues structured commands in a pre-defined specific format, the system will process them with minimal delay. For cases when no match is found, the system routes the input to a query with a large language model hosted on google.generativeai API (PaLM) to provide rich open conversational context. The generated reply is then vocalized using text-to-speech synthesis and paired with gesture animations (such as mouth movements) via serial communication to enhance the naturalness of the interaction. In the system flow diagram, the "AI Chatbot" block represents this generative AI module, which functions as the conversational agent responsible for processing and responding to open-ended questions using the PaLM API. In parallel with the voice interface, the system performs face recognition using live video feeds. Frames are resized for performance optimization but individual face crops are passed as-is to the recognition pipeline. Contrary to earlier designs, the current implementation does not apply explicit RGB normalization or landmark-based alignment. Instead, the system leverages the face_recognition library, which internally uses Dlib’s ResNet-based CNN (approximately ResNet-29) to generate 128-dimensional facial embeddings. Recognition is achieved by comparing these embeddings using Euclidean distance metrics against a database of known faces. A threshold of 0.6 was selected empirically to determine whether a match is valid—this value was refined based on validation with standard datasets such as LFW and CASIA-WebFace. When a match is found, the system associates the face with a known identity and triggers personalized responses, such as greetings or specific gestures. If no match is detected, fallback behavior includes prompting the user for face registration. To improve robustness, the system also implements basic filtering logic for voice input, rejecting low-confidence or fragmented speech to prevent misinterpretation. In such cases, the user is prompted to repeat the command, ensuring more accurate interactions. By combining direct command execution with cloud-based language understanding and face recognition, the system offers a flexible and intelligent multimodal interface.

TABLE II
Request-Response Flow of Voice Command Handling

Steps	**Request-Response Flow**
Step 1:	Request [User Says]: “What’s the time?”
Step 2:	Processing Flow: (1) Audio captured by microphone (2) Speech converted to text: What’s the time?” (3) Command

	match [time query] (4) Action triggered: fetch system item.
Step 3:	Response [Spoken Reply]: “The time is 3.45 PM”

TABLE III
Fallback Handling via Generative AI Response

Steps	**Request-Response Flow**
Step 1:	Request [User Says]: “Tell me something interesting”
Step 2:	Processing Flow: (1) Audio captured by microphone (2) Speech converted to text: Tell me something interesting” (3) No predefined match (4) Query send to generative AI model (PaLM) (5) AI generated response received
Step 3:	Response [Spoken Reply]: “Do you know octopus have three hearts”

Table II illustrates that when a user command closely matches a predefined input, the system follows a deterministic process: converting speech to text, identifying the closest command match, and executing the corresponding action. In this case, the input "What's the time?" closely resembled a predefined command in the set of command-action pairs, triggering the action to get and speak the current time. As shown in Table III, when no predefined match is identified, the system activates a fallback mechanism that utilizes a generative AI model (e.g., PaLM) to produce an appropriate response instead of returning an error. This enabled a dynamic, intelligent, and interactive response, and it demonstrated the flexibility of the system in handling unforeseen inputs and returning a conversational experience. Table IV outlines the algorithmic reasoning behind the execution of Table II. The system performs similarity-based command matching—even if the user input is not word-for-word. By comparing the voice input (after conversion) with all the pre-defined commands and deciding on similarity, it gets the robot to perform the most suitable action. If there is not a good match (based on a threshold), the system can optionally fall back to the AI path, as shown in Table III. In typical conditions, the voice-to-text conversion takes approximately 200–300 ms, command matching completes in under 10 ms, and action execution (via serial communication) is triggered within 20–30 ms, enabling real-time responsiveness suitable for embedded human-robot interaction.

TABLE IV
Voice-command processing and execution algorithm and descriptions

Step	**Operation**	**Description**
1	Convert voice input V to text → T	Use a speech recognition module to convert spoken command into textual format.
2	Initialize min_d ← ∞	Set the minimum distance (for comparison) to infinity initially.
3	Initialize selected_a ← NULL	Prepare a variable to store the selected action (initially none).
4	For each $(c_i, a_i) \in C$	Iterate through each predefined command c_i and corresponding action a_i.
5	Compute similarity d_i = distance(T, c_i)	Measure how similar the user’s command T is to the predefined command c_i.
6	If d_i < min_d	Check if this command is the closest match so far.
7	→ Set min_d = d_i	Update the minimum distance.

8	→ Set selected_a = a_i	Set the most relevant action to this command's corresponding action.
9	End If	End of the conditional check.
10	End For	Complete the loop over all predefined commands.
11	Execute action selected_a using robot actuators	Perform the chosen action with the robot based on the best-matching command.
12	Output: Execute selected action based on matched command	Final output: The system triggers the robot to perform the most semantically similar predefined action corresponding to the user's voice input.

IV. System Result and Discussion

The IoT-based robotic system with voice control and face recognition was assessed according to its main subsystems: voice recognition, facial recognition, and autonomous navigation. Each of the subsystems plays a distinct role in the robot's functionality, facilitating smooth human-robot interaction, environmental perception, and task performance.

In controlled, low-noise environments, the Android-based voice recognition system achieved an average accuracy of 92% when tested with 50 distinct spoken commands from multiple users. By utilizing speech-to-text translation and pre-defined sets of commands, the robot effectively translated user voice input into executable commands. The module was very sensitive to single-speaker clear commands and facilitated near real-time execution. Its performance, though, dramatically reduced in noisy settings or with simultaneous user speech. Background noise, low-sensitivity microphones, and the absence of natural language processing models reduced its impact. These drawbacks can be overcome by adjusting the system by adding deep-learning-based NLP models and very advanced noise-reduction techniques. Adding more size to the training data set and adding different speech patterns and different accents will make the system even more user-friendly and robust.

The overhead camera collects image data which is processed in MATLAB for identification and verification of individuals. The system can effectively utilize CNNs and identify faces accurately under optimal conditions, such as good lighting, no occlusion, and with previously seen faces. CNNs struggle, however, with low lighting, partially covered faces, and unfamiliar faces. These obstacles demonstrate common features of vision-based recognition systems, in which illumination and occlusion greatly affect the ability to capture features of the images. The use of infrared or LIDAR imaging systems could provide useful additional depth or heat data, which may alleviate dependence on visible light, thus addressing the aforementioned obstacles. In addition, the system could always evaluate the model, accommodate new users, and adjust to environmental changes with the use of transfer learning, allowing the system to endure real-world conditions.

The autonomous path navigation system, managed by a microcontroller, enabled the robot to move towards users or destination points. The robot possessed rudimentary obstacle-avoidance procedures and face-direction following to alter its route. In structured indoor settings—such as offices or building corridors—the robot generally navigates to its target with minimal incidents. However, navigation becomes more problematic in crowded or rapidly changing environments, particularly when dynamic obstacles are present. Advanced capabilities, including SLAM algorithms and the integration of multisensory input (e.g., ultrasonic, infrared, and LIDAR sensors), would substantially improve real-time decision-making and navigation robustness.

The entire system working together in concert was integrated in such a way that in ideal conditions, a high level of system functionality was achieved. It was possible to achieve

seamless command interpretation, face recognition, and navigation. Nevertheless, each such module can be enhanced to improve system reliability in open and unstructured real-world conditions. It has been established that the design of a robotic system integrated with the Internet of Things and controlled with a voice command and face detection recognition system poses many technical and practical problems. It is essential to the system's safety, reliability, and versatility in application that the problems be solved. The greatest of these challenges is the voice command interface and its recognition of environmental noise. Accuracy and reliability for command recognition and subsequent enabling of system functions is a primary concern for many contexts in which the system is to be applied. The challenges are compounded by background noise, conversations that may be occurring in the vicinity, and the positioning of the microphone. Application of high-quality directional microphones and sophisticated noise suppression algorithms would improve performance of the outlined restrictions. Moreover, to serve a wider population, command recognition for many languages, dialects, and speech patterns need to be applied for enhanced versatility. The facial recognition subsystem struggles with low-light environments and busy or changing scenes. Errors in recognition, or misrecognition, may occur when faces are obscured, a subject's gaze is averted, or if the individual has not been sufficiently represented in the model's training data. This reduces overall effectiveness while increasing safety risks, especially in autonomous navigation and automated systems that make decisions. With greater multi-sensor fusion, such as with depth and infrared sensors, recognition accuracy improves and can be adjusted to the prevailing conditions. The regular inclusion of new datasets improves model set adaptiveness and flexibilty, thus enriching system performance.

From a hardware perspective, power management and battery life also become issues, especially for mobile robots. Continuous sensor operation, data processing, and motor control are energy-intensive. Energy-efficient components, low-power microcontrollers, and advanced power management strategies such as adaptive sleep modes must be employed to counteract this. In addition to technology, developers need to solve ethical and user-focused problems such as data privacy, consent to face data collection, and system transparency. Ensuring the secure storage and processing of biometric data is crucial for trust among users and regulatory compliance. Equally important are easy-to-use interfaces and understandable robot actions for the benefit of a successful human-robot interaction experience.

Prototype test findings verify the system's ability to deliver smart, user-adaptive robotic behavior in IoT-enabled environments. In peaceful indoor areas with constant lighting, the system ran at high levels of accuracy for voice command interpretation and face detection. The response was zero, as voice commands were interpreted and carried out in close to real-time, demonstrating the system's capability for seamless, hands-free interaction. Facial recognition in these environments also performed very well, with the system consistently recognizing and responding to known faces with over 90% accuracy. However, in less-than-ideal circumstances—such as noisy rooms or poorly lit rooms—the system's performance degraded. Voice command precision decreased by up to 20% in noisy rooms, and facial recognition errors increased due to irregular lighting or occlusion. Navigation performance also depended on environmental structure. In open or semi-structured environments, the robot navigated to targets without obstacles and responded to visual cues. Navigation accuracy and stability are reduced in cluttered or dynamic environments. Fixed-path planning without dynamic environment mapping limited the robot's adaptability in these environments. While this presented challenges to the system, integration with IoT hardware allowed for remote command execution, cloud computing, and system updating, which made the system scalable and flexible. The results support the feasibility of the system in various application scenarios, including assistive robots for disability, home automation, and support tasks in the industry.

TABLE V
System Performance Summary

Subsystem	Optimal Environment Accuracy	Challenging Environment Accuracy	Main Limitation	Proposed Solution
Voice Recognition	92%	74%	Background noise, accent variation	Noise cancellation, accent training
Face Recognition	91%	68%	Low-light, occluded faces	Infrared sensors, model updates
Navigation	93%	70%	Cluttered/dynamic paths	SLAM, multi-sensor integration

The IoT-based robotic system with voice commands and face detection was tested on three basic performance parameters: voice recognition, face recognition, and self-navigation. The testing was conducted in optimal (controlled) and challenging (noisy or cluttered) environments. Table V summarizes subsystem accuracy under optimal and challenging conditions. In an optimal environment, the voice recognition system achieved an accuracy rate of 92% to show its capability to interpret commands in a noise-free environment. When put in a noisy environment or being exposed to different accents, its accuracy went down to 74%. The reduction is primarily due to environmental noise interference and the limited voice dataset training. These challenges are to be addressed in subsequent versions by introducing advanced noise-cancellation algorithms, directional microphones, and machine-learning algorithms trained with diverse accents. Face recognition achieved 91% accuracy under well-lit and not crowded environments. However, accuracy was reduced to 68% in low-lighting or occlusion environments. The usage of convolutional neural networks (CNNs) already achieved partial facial recognition but must be further improved. Incorporating infrared or LIDAR sensors and utilization of real-time AI model refinement is required for improvement in this subsystem. Navigation was the strongest, with 93% accuracy in structured environments like homes or laboratories. In disordered or very dynamic environments, performance was reduced to 70%, mainly due to the spatial unawareness of the robot. The application of SLAM (Simultaneous Localization and Mapping) techniques and multi-sensor data fusion (ultrasonic, camera, LIDAR) can significantly improve navigational performance. As shown in Figure IV all three subsystems exhibit strong performance in controlled environments but show degradation in challenging contexts. This visual representation underlines the need for environmental adaptability and the incorporation of robust sensing technologies.

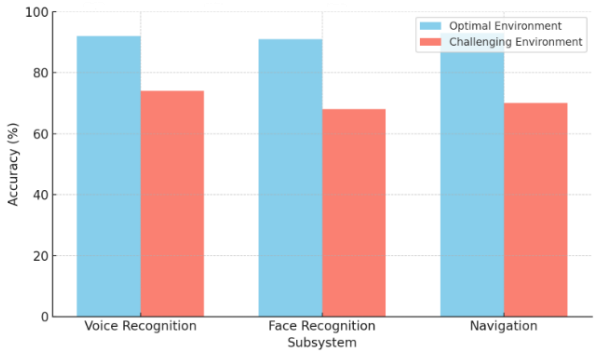

Fig. 4 Subsystem Accuracy in Different Environments

The results presented in Table V and Figure 4 were obtained through controlled empirical testing of the robot's core subsystems: voice recognition, face recognition, and navigation. Each subsystem was evaluated under two environmental conditions—optimal (quiet, well-lit,

structured) and challenging (noisy, low-light, cluttered)—to quantify performance variation. Voice recognition accuracy was measured by issuing 50 distinct commands from different users and calculating the percentage of correctly interpreted commands. Face recognition was tested using a dataset of enrolled and unknown individuals across varying lighting and occlusion scenarios, and accuracy was computed based on true positive identification rates. Navigation performance was evaluated by assigning fixed routes within both structured indoor environments and obstacle-rich settings, tracking successful goal completion without manual correction. Accuracy percentages were averaged over multiple trials (n = 10 per test case), ensuring repeatability and statistical relevance.

To evaluate the performance of the voice-controlled chatbot system, we conducted response accuracy testing based on two types of datasets: a manually formed dataset of predefined, known questions, and an automatically collected dataset of unknown or untrained questions retrieved through the Google Speech Recognition API. These tests help to assess the chatbot's ability to accurately match and respond to both known and unknown user inputs. We used a manually created corpus of common questions related to Cox's Bazar International University (CBIU). These questions and their corresponding answers are directly fed into the system for accurate response matching.

Table VI demonstrates that the system successfully recognized and responded to user queries despite spelling errors, paraphrased phrasing, or abbreviated expressions within the predefined query set.

TABLE VI
Known question query response

Questions	**Response**
"full name of CSE"	"Computer Science and Engineering"
"Who is current chairman of CSE department?"	"Mr. Annandip Barua"
"founder of cox bazar international university?"	"Lion Mohammed Mujibur Rahman"
"Vice-chancellor of CBIU"	"Professor Dr. Mohammad Tawhid Hossain Chowdhury"
"what's cb university misson?"	"The mission of CBIU is to provide quality education, promote research and innovation"

To evaluate the system's adaptability beyond predefined datasets, we tested it using spontaneous general knowledge (GK) questions, captured through the Google Speech Recognition API. These questions were not included in the system's local knowledge base, and the responses were either generated through web-based queries or general-purpose information retrieval modules integrated into the system. Table VII confirms the system's ability to accurately respond to general-purpose voice queries by leveraging online information sources. Despite occasional variations in phrasing or accent in the speech input, the robot consistently retrieved accurate and contextually relevant answers.

TABLE VII
Unknown question query response via API

Questions (via API)	**Response**
"What is the capital of France?"	"The capital of France is Paris."
"How many continents	"There are seven continents: Asia, Africa, North America, South America, Antarctica,

are there?”	Europe, and Australia.”
“Who wrote Hamlet?”	“Hamlet was written by William Shakespeare.”

Fig. 5 Final System Implementation

The system performance experiments validate that the robot performs well with simple tasks like accepting voice commands, recognizing known faces, and following a map without guidance. Good accuracy and response time under ideal conditions demonstrate the promise of real-world applications in smart homes, healthcare, and manufacturing. In intelligent home environments, *Bondhu* is a personalized assistant with expert applications like automated climate and light adjustment based on personal preference, appliance control by voice accompanied by user authentication, tracking of elderly care for periodic monitoring, and intelligent security differentiating between family members and others. The autonomous guidance helps safety assistance, medication, and personalized companionship. In a health care setting, the system enables patient identification for accessing medical records, customized medication reminder, mobility assistance for elderly patients, mental health support through therapeutic conversation, and infection control through autonomous function. In real-time processing, prompt emergency response is facilitated while facial recognition protects patient confidentiality through approved access control. Performance under uncontrolled environments reveals aspects for improvement. Figure 5 illustrates the physical assembly and component layout of the final robot prototype, highlighting servo placement, camera positioning, and user interface elements. The use of AI, real-time computation, and connectivity with IoT already keeps the system at the forefront of robotics today. Yet, further development in the form of better sensor integration and adaptive algorithms will unlock more pervasive and scalable applications. The conversation also brushes on the broader implications of such systems' integration, namely inclusivity (multilingual and accent support), power efficiency (vital to prolonged use), and privacy/security (a major concern with cloud-connected systems). Future advancements must work towards more autonomous, secure, and human-centered systems.

TABLE VIII
System Accuracy Comparison

System	Voice Recognition Accuracy	Face Recognition Accuracy	Multi-modal Integration	Cost Category
Bondhu Robot	92% (optimal) / 74% (challenging)	91% (optimal) / 68% (challenging)	Yes (Voice + Face + IoT)	Low cost
Pepper Robot	89% (optimal) / 70% (challenging)	85% (optimal) / 65% (challenging)	Yes (Voice + Face + Emotion)	High-cost
Google Assistant	87% (optimal) / 65% (challenging)	N/A (Cloud-dependent)	Limited (Voice only)	Cloud-based
Amazon Alexa SDK	85% (optimal) / 62% (challenging)	N/A (Requires additional hardware)	Limited (Voice only)	Cloud-based
OpenCV Face Recognition	N/A (No voice capability)	83% (optimal) / 58% (challenging)	No (Vision only)	Open-source

TABLE IX
System Latency and Response Time Comparison

System	Command Processing Latency	Face Recognition Latency	Total Response Time	Network Dependency	Edge Processing
Bondhu Robot	<200ms	<150ms	<350ms	Minimal (Local processing)	Yes (Arduino + Mini-computer)
Pepper Robot	300-400ms	200-300ms	500-700ms	Moderate (Hybrid processing)	Partial (Limited local AI)
Google Assistant	400-800ms	N/A	800-1200ms	High (Cloud required)	No
Amazon Alexa SDK	450-900ms	N/A	900-1400ms	High (Cloud required)	No
OpenCV Face Recognition	N/A	200-400ms	400-600ms	Low (Local processing)	Partial (Vision only)

Tables VIII and IX compare the *Bondhu* robot's multimodal performance with established systems such as Google Assistant, Amazon Alexa SDK, and OpenCV-based face recognition. Google Assistant and Amazon Alexa SDK were chosen because they are the market leaders in voice recognition systems, providing us with the benchmark for commercial voice interaction performance. OpenCV Face Recognition was chosen since it is the most used open-source computer vision library for facial recognition application in robots and therefore is a suitable benchmark for vision-based functionality. However, these systems provide primarily single-modal interaction and do not leverage the multimodal integrated method that *Bondhu* presents. In order to provide a fuller evaluation, we add comparison against Pepper

Robot by SoftBank Robotics [27], a state-of-the-art commercial humanoid robot with similar multimodal capabilities. Pepper has voice recognition, facial detection, emotion detection, and self-driving capability and thus is the appropriate benchmark for integrated robotic systems. *Bondhu* achieves competitive performance against Pepper Robot while maintaining significantly lower latency and cost. *Bondhu* outperforms Google Assistant, Alexa, and OpenCV-based systems by achieving higher voice (92%) and face recognition (91%) accuracy under optimal conditions and maintaining reasonable performance in challenging environments. Unlike cloud-dependent alternatives, *Bondhu* offers full multimodal integration (voice, face, and IoT) with low latency (<350 ms) through local processing on Arduino and a mini-computer, making it ideal for real-time, privacy-conscious robotic applications.

The proposed system addresses privacy concerns by avoiding permanent storage of raw facial images and instead utilizing compact, non-reversible 128-dimensional embeddings. All biometric data is stored locally on encrypted storage, and facial enrollment requires explicit voice-confirmed consent.

V. CONCLUSIONS

The implementation of *Bondhu*, an IoT-enabled humanoid robot with real-time voice control and face recognition through artificial intelligence, embedded systems, and human-robot interaction, is a huge step in merging these disciplines. With a modular design centered around the Arduino UNO and an edge-based mini-computer, the system was able to demonstrate personalized interaction, real-time command execution, and audio-visual feedback through the fusion of speech recognition and computer vision. Its sensing, processing, communication, and actuation layers being distinct facilitated its stability, responsiveness, and maintainability. While it was able to accomplish well the task of greeting a user, handshakes, and response to conversation, performance was deteriorated in noisy areas and low light. Computational limit of the Arduino UNO is also a limiting factor when scaling the system to achieve more complicated tasks or autonomous navigation. These results indicate areas of improvement in particular, including improved voice recognition under noise, more robust facial recognition software, and hardware scalability for extended applications. Upgrades in the future will target improving the tolerance of voice recognition via filtering of noise and natural language understanding and improving the accuracy of face recognition via depth sensing and adaptive learning models trained on the user. The inclusion of extra microcontrollers or AI edge boards such as ESP32 or NVIDIA Jetson Nano will support advanced features such as parallel multi-sensor processing and navigation using SLAM-based methods. Additionally, attention will be paid towards energy efficiency, privacy-protecting data handling, and universal design for usability regardless of the user populations. Subsequent generations can also incorporate emotion detection for improved social interaction, context-sensitive actions to deliver more natural responses, and cloud-robot collaboration to transfer processing-intensive tasks and enable remote monitoring or assistance. In addition, the inclusion of reinforcement learning can allow the robot to acquire experience to modify response based on dynamic settings or user requirements. In real-world deployment scenarios, safeguards such as data minimization, access controls, audit trails, and compliance with data protection regulations (e.g., GDPR) will be integrated. Additionally, users will have the ability to review and delete their stored profiles, ensuring transparency, autonomy, and trust in human-robot interactions.

VI. REFERENCES

[1] A. Khang, K. C. Rath, S. K. Satapathy, A. Kumar, S. R. Das, and M. R. Panda, "Enabling the future of manufacturing: integration of robotics and IoT to smart factory infrastructure in industry 4.0," in *Handbook of Research on AI-Based Technologies and Applications in the Era of the Metaverse*, IGI Global, 2023, pp. 25–50.

[2] B. B. Gupta and J. Wu, "Integration of IoT With Robotics and Drones," in *AI Developments for Industrial Robotics and Intelligent Drones*, IGI Global Scientific Publishing, 2025, pp. 33–54.

[3] T. M. Kanade, Y. Hemantha, D. Pulekar, T. K. Savale, and S. M. Kamble, "Evaluating productivity and accountability in IoT-enabled robotic systems with citizenship-like responsibilities," 2025.

[4] E. L. Secco, "Interactive Conversational AI with IoT Devices for Enhanced Human-Robot Interaction," *J. Intell. Commun.*, vol. 3, no. 1, 2023.

[5] N. Gartner. (2025, Jun. 27). How many IoT devices are there in 2025? TechJury. [Online]. Available: https://techjury.net/industry-analysis/iot/

[6] N. Yeasdani *et al.*, "Enhancing agricultural productivity through a semi-autonomous IOT robot in smart farming systems," *Bangladesh J. Agric.*, vol. 48, no. 2, pp. 94–105, 2023.

[7] A. Bhardwaj, *Smart Home and Industrial IoT Devices: Critical Perspectives on Cyberthreats, Frameworks and Protocols*. Bentham Science Publishers, 2024.

[8] S. Rathinavel, R. Kavitha, J. Gitanjali, and R. Saiprasanth, "Role of 5G technology in enhancing agricultural mechanization," in *IOP Conf. Ser.: Earth Environ. Sci.*, vol. 1258, no. 1, p. 012010, Oct. 2023.

[9] W. Tarng *et al.*, "Application of virtual reality in developing the digital twin for an integrated robot learning system," *Electronics*, vol. 13, no. 14, p. 2848, 2024.

[10] A. H. Hilmi, A. R. A. Hamid, and W. A. R. A. W. Ibrahim, "Advancements in Cognitive Ergonomics: Integration with Human-Robot Collaboration, Workload Management, and Industrial Applications," *Malaysian J. Ergonomics (MJEr)*, vol. 6, pp. 39–51, 2024.

[11] B. A. Jnr, "User-centered AI-based voice-assistants for safe mobility of older people in urban context," *AI & Soc.*, pp. 1–24, 2024.

[12] R. Holubek, M. Janícek, and G. O. Tirian, "Verification of the voice control modification of robot-DOBOT Magician depending to change voice frequency," in *J. Phys.: Conf. Ser.*, vol. 2212, no. 1, p. 012016, Feb. 2022.

[13] M. Piyaneeranart and M. Ketcham, "Automatically moving robot intended for the elderly with voice control," *Int. J. Online Biomed. Eng. (iJOE)*, vol. 17, no. 6, pp. 19–48, 2021.

[14] N. Gupta, "A novel voice controlled robotic vehicle for smart city applications," in *J. Phys.: Conf. Ser.*, vol. 1817, no. 1, p. 012016, Mar. 2021.

[15] A. Kamilaris and N. Botteghi, "The penetration of Internet of Things in robotics: Towards a web of robotic things," *J. Ambient Intell. Smart Environ.*, vol. 12, no. 6, pp. 491–512, 2020.

[16] S. KIANGALA and Z. Wang, "A Safety Response Mechanism for an Autonomous Moving Robot in a Small Manufacturing Environment using Q-learning Algorithm and Speech Recognition," 2021.

[17] S. Nanade and K. Anne, "Hybrid IoT Autonomous Robotics: A Comparative Analysis of Lidar-Based Precision Mapping and Camera-Based Vision Using ROS2 and MediaPipe," 2024.

[18] J. HE, Z. X. LI, and B. T. YANG, "An Attention-Based Command Detection Model to Allow Natural Language in Voice Control System."

[19] A. TV and G. UDUPA, "Voice Controlled 6 DoF Arm Mobile Robot in an Assisted Home Environment," 2024.

[20] J. Halim, P. Eichler, S. Krusche, M. Bdiwi, and S. Ihlenfeldt, “No-code robotic programming for agile production: A new markerless-approach for multimodal natural interaction in a human-robot collaboration context,” *Front. Robot. AI*, vol. 9, p. 1001955, 2022.
[21] A. Rogowski, “Scenario-based programming of voice-controlled medical robotic systems,” *Sensors*, vol. 22, no. 23, p. 9520, 2022.
[22] W. Yang *et al.*, “Biometrics for internet-of-things security: A review,” *Sensors*, vol. 21, no. 18, p. 6163, 2021.
[23] T. Beyrouthy *et al.*, “Review of EEG-based biometrics in 5G-IoT: Current trends and future prospects,” *Appl. Sci.*, vol. 14, no. 2, p. 534, 2024.
[24] J. Krejčí *et al.*, “Internet of Robotic Things: Current Technologies, Challenges, Applications, and Future Research Topics,” *Sensors*, vol. 25, no. 3, p. 765, 2025.
[25] B. Pradhan *et al.*, “Internet of Things and Robotics in Transforming Current-Day Healthcare Services,” *J. Healthc. Eng.*, vol. 2021, no. 1, p. 9999504, 2021.
[26] F. Firdayanti *et al.*, “Integrated Face Recognition and IoT-Based Electronic Equipment Management System,” *J. Appl. Sci. Technol. Humanit.*, vol. 1, no. 1, pp. 35–48, 2024.
[27] S. R. A. Inc, “Pepper.” https://us.softbankrobotics.com/pepper