ECE 445

SENIOR DESIGN LABORATORY

PROJECT PROPOSAL

Visual Chatting and Real-Time Acting Robot

Team #37

HAOZHE CHI
(haozhe4@illinois.edu)
MINGHUA YANG
(minghua3@illinois.edu)
JIATONG LI
(jl180@illinois.edu)
ZONGHAI JING
(zonghai2@illinois.edu)

TA: Enxin Song

March 26, 2024

1 Introduction

1.1 Problem Statement

Blind individuals often face significant difficulties when navigating unfamiliar environments, such as finding water dispensers in large public spaces. Additionally, there is a risk of injury from interacting with devices that dispense hot water. The emergence of large language models (LLMs) and large visual language models (LVLMs) offers a promising avenue for developing innovative solutions to these challenges.

1.2 Solution Overview

We propose to create an AI-enhanced robotic service system designed to assist blind people by guiding them to water dispensers, providing real-time vocal instructions for navigation and safety, and autonomously refilling water bottles. This system will integrate a camera mounted on the user's head for visual input and use speech-to-text AI technology to interpret verbal commands. The core of our solution is the BLIP-2 visual language AI model, which will process both visual and textual inputs to generate actionable guidance. Additionally, a text-to-speech AI will transform text outputs into auditory instructions, thereby facilitating the user's interaction with their environment. Our system comprises the following key components:

- **Real-Time Visual and Verbal Input Processing:** A combination of a head-mounted camera and speech-to-text AI captures and analyzes the user's surroundings and voice commands.
- **Dynamic Guidance and Interaction:** The BLIP-2 model will provide navigation assistance, warn of potential dangers, and instruct on interacting with a water dispenser.
- **Autonomous Assistance:** A Universal Robot Arm UR3e, controlled by the Robot Operating System and instructed by the Vision Language AI model, will autonomously refill the user's water bottle.
- **User Communication:** Audio feedback and instructions will be delivered through a Bluetooth headset, ensuring clear and effective communication.

Operational Process When a blind individual approaches a water dispenser, the system triggers a specific sequence of actions:

1. The Vision Language AI model guides the user to sit and place their water bottle in a designated location.
2. Subsequently, a robot arm, following instructions from the Vision Language AI model, securely grasps the bottle, fills it with water from the dispenser, and then returns the filled bottle to the user.

Challenges and Innovation Integrating LVLMs within a robotic framework presents unique technical challenges, especially in accurately translating AI-generated instructions into precise robotic movements. This project not only aims to address a tangible issue but also seeks to advance the field of AI and robotics by exploring new applications of visual language models in assistive technology.

1.3 Visual Aid

The visual illustration of our AI-enhanced robotic service system is shown in Figure 1.

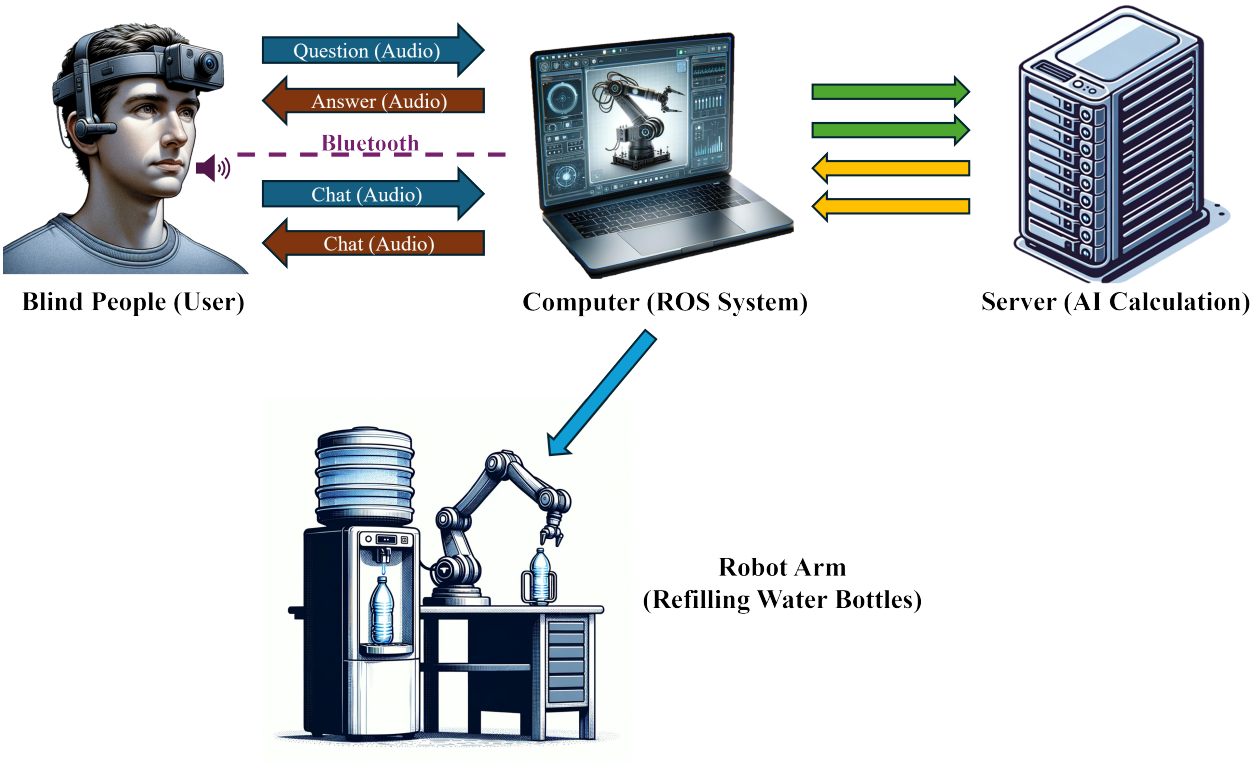

Figure 1: Visual Illustration of the AI-enhanced Robotic Service System

1.4 High-level Requirements List

1. **Server Specifications:** A high-performance server equipped with a GPU boasting a minimum of 24 GB of memory is essential for efficiently running the Visual Language AI model.
2. **Robotic Arm:** The project demands a robotic arm with a minimum of six joints for versatile movement and object manipulation capabilities. An example of such hardware is the Universal Robot UR3e.

3. **Camera System:** We require two cameras capable of capturing real-time images. The primary camera will provide visual feedback from the perspective of blind users, while the secondary camera will be used to detect the presence and position of water bottles. Both cameras must be compatible with the Robot Operating System (ROS), such as the CK camera model.

2 Design

2.1 Block Diagram

The overall block diagram of our AI-enhanced robotic service system is shown in Figure 2.

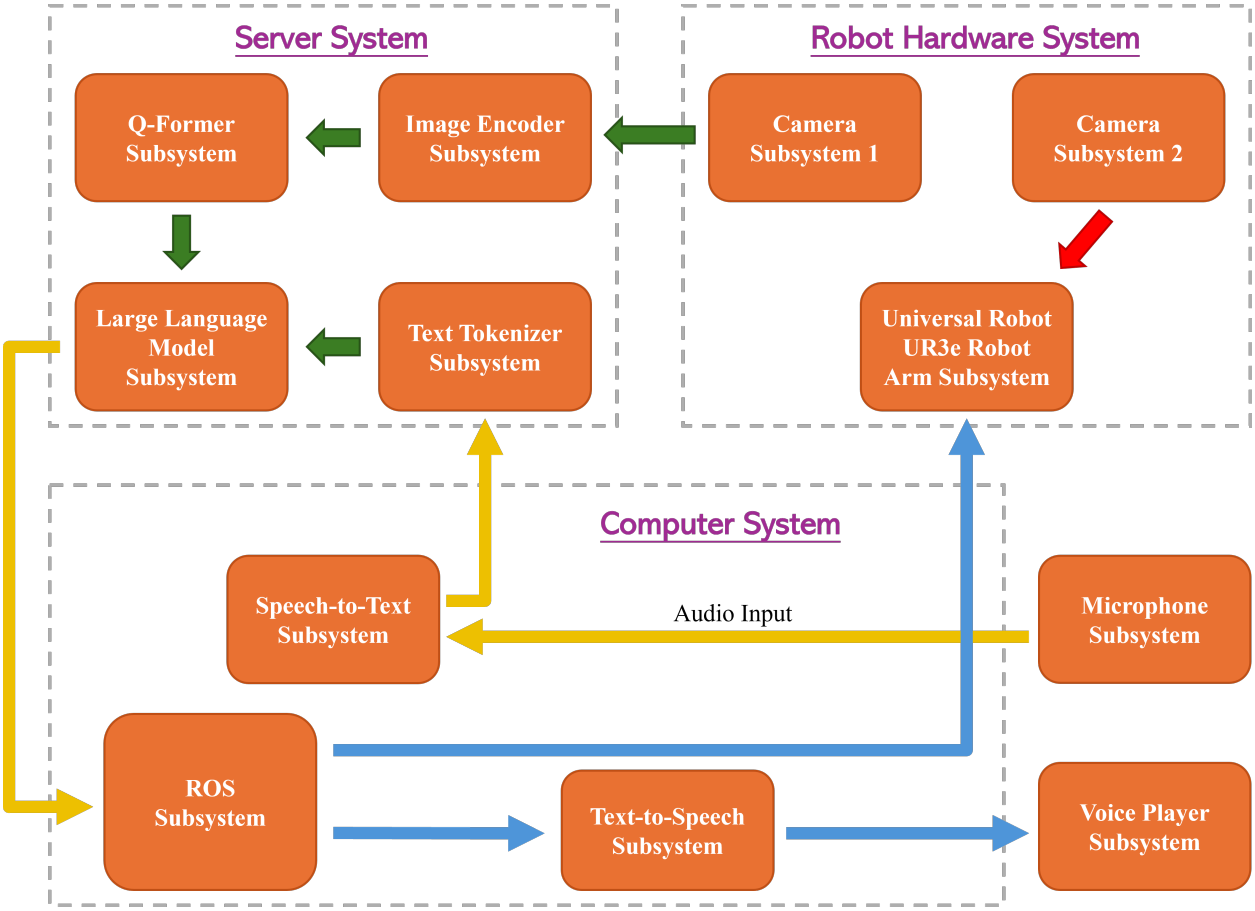

Figure 2: Block diagram of the AI-enhanced robotic service system (Green arrow: visual flow; Yellow arrow: text flow; Red arrow: attach itself to; Blue arrow: instruction flow), in which Camera Subsystem 1 is for blind people and Camera Subsystem 2 is for robot arm.

2.2 Subsystem Overview

Q-Former Subsystem The Q-Former subsystem, based on transformer architecture, transforms visual features into more abstract representations by incorporating attention mechanisms. It receives visual features from the Image Encoder subsystem and relays enhanced features to the Large Language Model subsystem for further processing.

Image Encoder Subsystem Utilizing a Vision Transformer (ViT) structure, this subsystem encodes images into visual features for analysis. It captures images from the Camera subsystem and forwards the processed visual features to the Q-Former subsystem.

Text Tokenizer Subsystem This subsystem tokenizes and converts input text into embeddings, facilitating textual analysis. Although it is mentioned that it receives images from the Camera subsystem, it should correctly receive text inputs, likely from the Speech-to-Text subsystem, and send the resulting embeddings to the Large Language Model subsystem.

Large Language Model Subsystem Processing both visual and textual embeddings, the Large Language Model subsystem generates text outputs that encompass instructions for robotic actions and responses to queries. These outputs are directed to the ROS subsystem for action and response articulation.

Camera Subsystem Operating continuously, the Camera subsystem captures images every second, providing real-time visual data to the Image Encoder subsystem for processing.

Universal Robot UR3e Robot Arm Subsystem Receiving instructions from the ROS subsystem, the UR3e robot arm picks up the water bottle from a designated area, moves it to the water dispenser for filling, and then returns the filled bottle to its original location following further ROS instructions.

Speech-to-Text Subsystem This subsystem converts spoken language into textual format, supplying the Text Tokenizer subsystem with text for embedding and further processing.

ROS Subsystem As the control hub, the ROS subsystem manages hardware operations based on instructions and responses generated by the Large Language Model subsystem. It orchestrates the execution of robotic actions and relays responses to the Text-to-Speech subsystem.

Text-to-Speech Subsystem Converting textual responses into audible speech, this subsystem ensures the robot can verbally communicate with users, passing audio outputs to the Voice Player subsystem for playback.

Voice Player Subsystem This subsystem is responsible for audibly playing back responses generated by the Text-to-Speech subsystem, enabling interactive communication between the robot and its human users.

Microphone Subsystem Captures audio from users, sending it to the Speech-to-Text subsystem.

Raspberry Pi Auxiliary Subsystem The Raspberry Pi Auxiliary System serves as a monitoring and guidance mechanism to facilitate interaction between the user and the Robot Arm. It is equipped with two cameras and a speaker. One camera is tasked with detecting the placement of the water bottle on the desk, ensuring it is within the Robot Arm's reach. The second camera confirms the bottle's presence at the water dispenser. Upon detecting a misplaced bottle, the system issues vocal instructions via the speaker to aid the user in correctly positioning the bottle.

PCB Water Dispenser Subsystem The PCB Water Dispenser Subsystem is designed to visually signal the operational status of the water dispenser using a simple light control mechanism. Upon receiving a signal from the Raspberry Pi Auxiliary System that a water bottle is in position, the PCB activates a green light to signify the dispensing of water. Following a 5-second fill time, the light transitions to red, indicating the completion of the process. This change in status is communicated back to the Raspberry Pi Auxiliary System, prompting the Robot Arm to retrieve the bottle and return it to the user.

2.3 Subsystem Requirements

Q-Former Subsystem The Q-Former subsystem acts as an integrative bridge, merging the capabilities of a static image encoder with a static Large Language Model (LLM). It is designed to extract and refine output features from the image encoder, irrespective of the input image's resolution. This subsystem comprises two transformer submodules utilizing shared self-attention layers, facilitating efficient processing of visual features and textual data. The image transformer submodule is dedicated to processing visual information received from the image encoder, while the text transformer submodule functions dually as an encoder and a decoder for textual information. This dual functionality enables the Q-Former to seamlessly integrate visual and textual inputs for comprehensive processing.

Image Encoder Subsystem The Image Encoder subsystem employs a transformer-based architecture, specifically a **ViT-L/14** model, to encode images into a compact, feature-rich representation. It utilizes 32 distinct queries, each with a dimension of 768, matching the Q-Former's hidden dimension. The output, denoted as Z, adopts a 32×768 dimension, significantly reducing the representation size compared to the original image features extracted by the ViT-L/14 (257×1024), thus enhancing processing efficiency.

Text Tokenizer Subsystem The Text Tokenizer subsystem is responsible for converting raw text input into structured embeddings, using a tokenizer consistent with the **BERT** model. This approach ensures compatibility with widely utilized Large Language Models, such as Llama, facilitating seamless integration within the broader architecture and promoting effective textual analysis.

Large Language Model Subsystem This subsystem leverages an open-source Large Language Model, such as Llama, serving as a downstream decoder to process the combined input features from both visual and textual sources. The choice of **Llama** aligns with the architectural principles of BLIP-2 [1] based systems, enabling sophisticated text output generation that includes both instructions for robotic actions and responses to user inquiries.

Camera Subsystem A head-mounted camera captures the user's surroundings and sends the images to the Large Language Model Subsystem, which then give instructions to the user how to reach the water dispenser.

Universal Robot UR3e Robot Arm Subsystem Utilizing the Universal Robot UR3e, this subsystem is central to the robot's physical interactions with its environment. The UR3e is chosen for its flexibility and precision, featuring six joints that facilitate a wide range of movements and tasks.

Speech-to-Text Subsystem This subsystem employs an open-source model, such as those available from Google, to convert human speech into text. The processed textual data is then relayed to the Text Tokenizer subsystem, ensuring that voice commands are accurately interpreted and acted upon.

ROS Subsystem The Robot Operating System (ROS) subsystem serves as the robot's meta-operating system, offering essential services including hardware abstraction, device control, and inter-process communication. ROS's comprehensive ecosystem supports the development and execution of robotic applications across diverse hardware setups.

Text-to-Speech Subsystem Employing ***pyttsx3***[1], an open-source Python library for offline text-to-speech conversion, this subsystem transforms text outputs from the ROS subsystem into audible speech. This allows the robot to communicate responses to user queries verbally, enhancing the interactive experience.

Voice Player Subsystem The Voice player subsystem is responsible for the audible output of the robot's responses. It receives audio files from the Text-to-speech subsystem and plays them out loud, enabling the robot to communicate effectively with its human users. Needs to incorporate a Bluetooth headset for clear audio playback.

[1]https://pyttsx3.readthedocs.io/en/latest/

Microphone Subsystem Should use a Bluetooth microphone for flexible and reliable audio capture from users.

Raspberry Pi Auxiliary Subsystem Each camera should be positioned to oversee the water bottle's placement both on the desk and at the water dispenser. A speaker, integral to the system, is responsible for delivering clear and audible voice instructions to direct the user. This is supported by sophisticated object recognition model, designed to accurately determine the bottle's location and offer precise feedback. At its core, the system is built to provide user-centric feedback, generating straightforward vocal guidance such as "Please move your bottle left about 10cm," thus ensuring the user can easily follow the instructions provided.

PCB Water Dispenser Subsystem An interface for receiving placement confirmation signals from the Raspberry Pi Auxiliary System, programmable LEDs or equivalent light indicators to display the dispenser's status via color changes from green to red, an internal timer to regulate the duration of the dispensing process and manage the light transitions, and the capability to send a completion signal back to the Raspberry Pi Auxiliary System, which in turn prompts the Robot Arm to initiate the retrieval of the water bottle.

2.4 Tolerance Analysis

A key design consideration is the latency in data transfer, which is critical to real-time interaction and control. We meticulously assess the latency focusiing on two main channels: user to computer, and computer to server.

User to Computer Data Transfer Analysis: A pivotal design concern is the latency during Bluetooth transmission of captured images and audio from head-mounted cameras and headsets to the computer. Assuming an operational distance of approximately 10 meters, we utilize the following formula to estimate Bluetooth transmission latency:

$$\text{Latency} = \frac{\text{Data Size}}{\text{Transmission Speed}} + \text{Propagation Delay}$$

Data Size is the total size of the data to be transmitted, measured in bits. Transmission Speed is the rate at which data is transmitted, measured in bits per second (bps). Propagation Delay is the time it takes for the signal to travel from the source to the destination, which can be calculated as the distance divided by the speed of the signal. However, for Bluetooth and similar short-range technologies operating at the speed of light, this delay is negligible compared to other factors.

Given Bluetooth 4.0's capability of up to 25 Mbps in high-speed mode and considering an average data packet size (1MB for a captured image), we can estimate the latency:

$$\text{Latency} = \frac{1 \times 10^6 \times 8 \,\text{bits}}{25 \times 10^6 \,\text{bits/sec}} = 0.32 \,\text{seconds}$$

Computer to Server Data Transfer Analysis: Through simulations, we have estimated that data transfer delays between the computer and server can be confined to approximately 3-4 seconds. This latency is primarily influenced by network speed, server processing capabilities, and the data's complexity. Incorporating Python libraries like flash-attention has been instrumental in augmenting our AI models' processing speeds. These libraries enable more efficient handling of computations necessary for real-time analysis and decision-making based on the data received from user devices.

Conclusion: Experimental outcomes demonstrate that, despite variations, the entire processing duration stays within a few seconds, contingent on the complexity of the input data. This duration falls within our acceptable limits for real-time operations, underscoring the system's viability for responsive and effective user assistance. This analysis confirms our commitment to optimizing system performance while maintaining the real-time interaction that is vital for the success of our project.

3 Ethics and Safety

In the development of our robotics project, we rigorously adhere to the IEEE Code of Ethics [2] to uphold the highest standards of ethical practice and safety.

3.1 Ethics

Privacy (ACM 1.7: Respect the Privacy of Others) To protect privacy, we take strict measures to ensure the confidentiality and security of any personal information collected by the robot. We establish robust protocols based on industry standards to limit access to this sensitive data to authorized individuals with a legitimate need to access it. In addition, we carefully design and implement secure storage and processing procedures to reduce the risk of unauthorized disclosure or misuse. By prioritizing the protection of personal information, we demonstrate our unwavering commitment to maintaining the privacy and trust of individuals who interact with our robots.

Fairness (IEEE - Avoiding Real or Perceived Conflicts of Interest) The possibility of bias in the decision-making process of artificial intelligence is a major ethical issue. Recognizing this, we will strive to provide robots with a comprehensive understanding of human diversity and societal nuances through rigorous training and careful refinement. By exposing robots to a variety of data, including different demographics, cultural backgrounds, and environmental scenarios, we aim to equip robots with the ability to impartially discern and understand complex social dynamics.

Being Open (ACM 1.2: Avoid Harm) We are committed to ensuring full transparency in the robotics decision-making process. Our goal is to provide clear and understandable information to all stakeholders so that they can fully understand how the robot operates and the factors that influence its decisions. To achieve this, we keep detailed records of

the algorithms, data inputs and learning methods used by the robot. Additionally, we are committed to an open approach to making information about the robot's functioning, including its training data, learning outcomes, and decision logic, readily available. By increasing transparency, we aim to build trust and confidence among users, stakeholders, and the broader community, thereby promoting ethical behavior by individuals or organizations when using our robotics.

Professional Development (ACM 2.6) Adhering to ACM's principles, our team dedicates itself to the continual enhancement of our knowledge and understanding of the societal ramifications of robotics. We recognize the dynamic nature of ethical standards and proactively refine our systems to stay abreast of new developments, ensuring that our robots serve as a benchmark for responsible AI and robotics practice.

3.2 Safety

Avoiding Accidents (IEEE - Priority to Public Welfare) Our robots are carefully designed with safety as a top priority to ensure that they do not jeopardize the personal safety of others or the safety of property. Equipped with advanced emergency stops and a range of sophisticated sensors, the robots are able to operate with increased vigilance, effectively preventing collisions with people and objects. These safety features are carefully designed to prevent accidental collisions and provide peace of mind in dynamic environments where human-robot interactions are frequent.

Staying Secure (ACM 3.7: Recognize the Need to Protect Personal Data) Given the advanced functionality and interconnectedness of our robots, it is critical to protect their integrity and guard against potential cyber threats. We are therefore building relevant security measures to strengthen its defenses and reduce the risks posed by malicious actors and cyberattacks. This requires the implementation of advanced encryption protocols, strict access controls and continuous monitoring mechanisms to detect and respond to any unauthorized attempts to compromise robotic systems or data. In addition, we prioritize regular security assessments and audits to identify vulnerabilities and weaknesses in our security infrastructure, enabling us to proactively address potential threats and ensure that our robots are resilient to evolving cyber threats.

Dealing with Mistakes (ACM 2.5 & IEEE - Acknowledge and Correct Mistakes) In the event of an unforeseen situation or error, the robot responds in a manner that prioritizes safety and reliability. The robot's operational framework incorporates fail-safe mechanisms and real-time monitoring capabilities to promptly identify and address any anomalies or deviations from expected behavior. By promptly notifying designated personnel or stakeholders of such occurrences, the robot facilitates rapid intervention to minimize potential risks and ensure continuity of safe and effective operations.

Responsibility (IEEE) In line with IEEE guidelines, our project is committed to the responsible deployment of robotics, ensuring they fulfill their intended roles effectively

while safeguarding societal and environmental well-being. Our team maintains a vigilant approach to technology stewardship, regularly assessing and mitigating any negative impacts our robots may have, thereby ensuring our innovations contribute positively to society and operate sustainably within the environment.

Whistleblowing (ACM 1.4) Upholding ACM's ethical code, we foster an environment where whistleblowing is not just protected but encouraged, as it is crucial for maintaining the highest ethical standards. By promoting transparency and inviting scrutiny, we ensure any instance of misuse or ethical misconduct involving our robots is promptly addressed, reinforcing our commitment to integrity and the responsible use of technology.

References

[1] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," *arXiv preprint arXiv:2301.12597*, 2023.

[2] Institute of Electrical and Electronics Engineers. (2016) IEEE Code of Ethics. Accessed on: 2024-03-07. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html