ECE 445

SENIOR DESIGN LABORATORY

EARLY PROPOSAL

Early Proposal for ECE 445

Team #

CHENGHAN LI (cli104@illinois.edu)
YIPU LIAO (yipul2@illinois.edu)
YIMING LI (yiming20@illinois.edu)
HAORAN YANG
(haorany8@illinois.edu)

TA: Leixin Chang

March 7, 2024

Contents

1 Introduction

1.1 Problem

With the intensification of China's aging population, diseases of old age are becoming more and more serious in China, and the number of patients is increasing rapidly. Among them, Parkinson's disease (PD) is a common neurological degenerative disease, with an average onset age of 60 years old. The prevalence of PD in people over 65 years old in China is about 1.7%. With the increase of age, this prevalence rate further increases to more than 4% for elderly over 80 years old. By 2030, the number of Parkinson's disease patients in China will reach 5 million, accounting for almost half of the global number of patients.
Patients with Parkinson's disease may experience tremor, rigidity, bradykinesia, or postural gait disturbance, which can seriously affect their daily life. We often see Parkinson's patients can't freely control the body to complete the removal of objects and other actions. so, a simple operation, intelligent function of the health assistance and daily cooperation system is very important.

1.2 Solution

Our graduation project aims to conceptualize an intelligent robotic arm capable of proficiently performing various tasks through speech and visual recognition. The most important concept is that the user verbally identifies objects on the table to the robot. After receiving the voice command, the robot arm determines the target, uses its camera and visual recognition system to detect and locate the target object, and then carries out corresponding operations on it, such as picking up the target object and placing it in the required position.
The whole system is shown in 1. The specific system can be divided into three subsystems: language model, visual recognition, and robotic arm. These systems interact with each other and cooperate to complete the order. First, the language model recognizes the operator's language instructions and converts them into computer signal instructions for transmission to the visual recognition system. The visual recognition system automatically recognizes the target object according to the instruction and the visual information from the camera as input, and generates the position information of the target object as output. Finally, after the robot arm obtains the position information from the visual recognition system, the corresponding motors move initially, reaches the predetermined position and completes the command task.

The Overview of the whole system1. The left subsystem is the audio recognition subsystem, it contains a microphone and an audio recognition network. The middle subsystem is the visual detection subsystem, it contains a visual detection network and a RealSense depth camera. The right one is the robot arm subsystem, it contains our ROS system on Ubuntu and a set of robotic arm and mechanical structure.

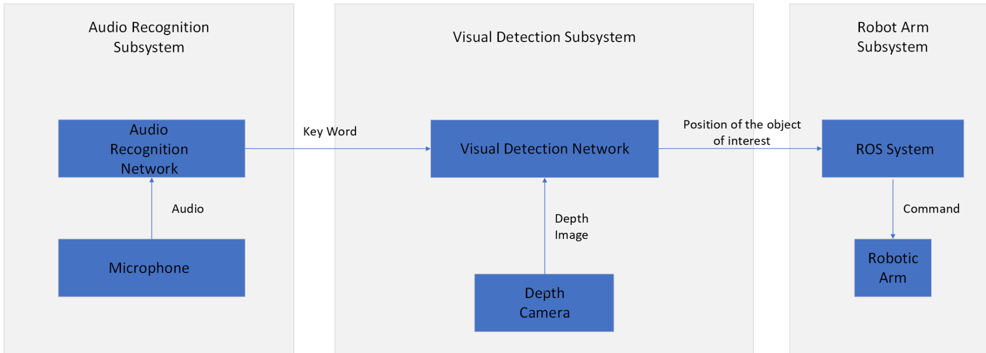

Figure 1: The Overview of the whole system

1.3 High-level Requirements List

Reliability: The system maintains a high level of reliability and recognition accuracy, and our goal is to maintain the accuracy of speech recognition and visual recognition above 90%.
Intelligence: The robot arm should be designed to recognize and execute enough commands to complete them. It can understand human natural language, and carry out relational extraction to extract the object of the sentence accurately.

2 Design

2.1 Block Diagram

The overall design process involves inputting both voice and image data into the system. The system then recognizes and processes these inputs, before sending feedback to the robotic arm, instructing it to perform corresponding actions. This process encompasses the complete journey from perception to execution, achieving the goal of intelligently controlling the robotic arm's operations through voice and image inputs.

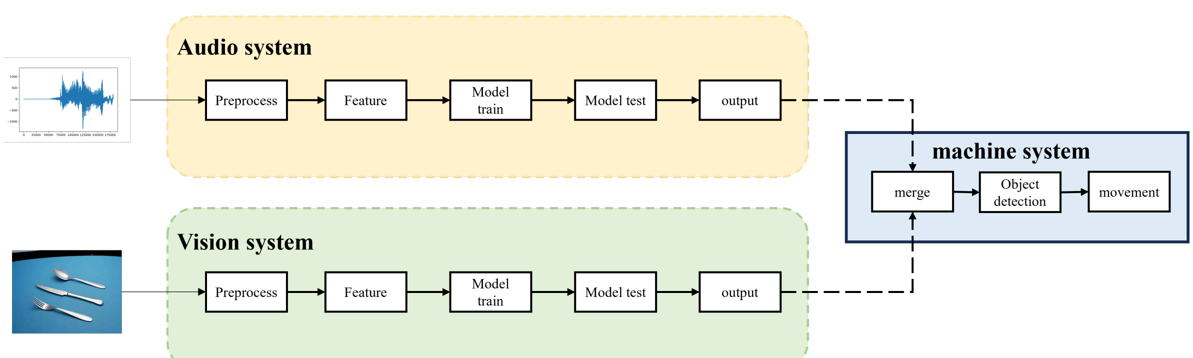

Figure 2: Block diagram

2.2 Subsystem 1: Robot Arm

2.2.1 Overview

Robotics arm system This subsystem contains an arm and grab mechanism. We plan to use 4-axis arm, which means there are 4 motors or joints to control the arm. This subsystem will receive the target point coordinates of the object from the vision system and use the Inverse Kinematics Algorithm to calculate the rotation of each joint. This algorithm will be run on a PC inside of a signal chip microcomputer. For the grab mechanism, we will use a clamp to take objects. Since all the objects we will grab have some hardness, we do not need to consider adding a sensor to the clamp to control the force that will be applied to the object.

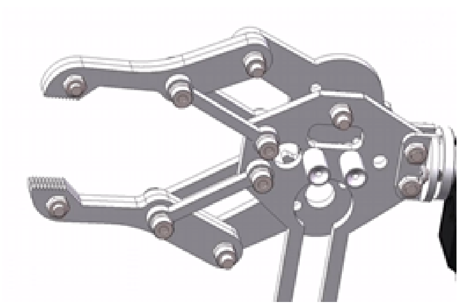

Figure 3: End-effector

Figure 4: Robotics arm

For the material, we will use PLA and use 3D printing technology to print all parts For the arm joints, we plan to use Xiaomi's Cyber Gear motors and KP035 bearing to achieve the movement of the robotics arm.

2.2.2 Requirements

Range: The workspace of robotics arm is shown below 5 6
payload: For this robotics arm, the max load on the end-effector is limited to 3kg.
Operating voltage: power supply should be capable of supplying 24V and 15A or higher.
Motor type: Brushless DC Servo(H54P Series), Coreless DC Motor(H42P Series)
Communication Baudrate: 1000000 bps

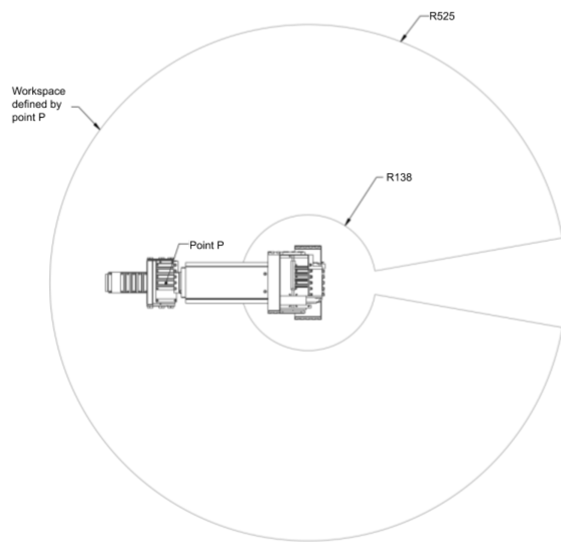

Figure 5: Workspace top view

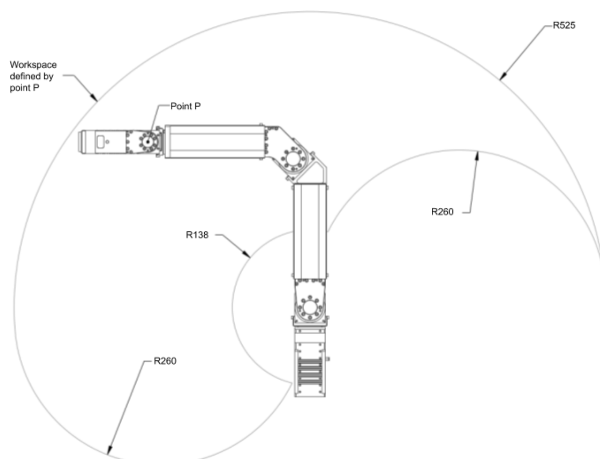

Figure 6: Workspace side view

2.3 Subsystem 2: Visual Model

2.3.1 Overview

Our task is to perform real-time single-object detection using the camera mounted on the robotic arm. Therefore, we need to employ a lightweight and fast single-object detection algorithm and network. This choice ensures that we can effectively identify and locate target objects while meeting real-time requirements. During the selection process, we will prioritize algorithms and networks with lightweight characteristics to ensure optimal performance within limited resources. The input of this subsystem is the output label (object) of the language model, and the image obtained from the camera. We consider our task as a traditional single object detection task; the output of this task should be the relative location of the input object in the input image. Then according to the relative

location, we estimate the location of the object in the camera coordinate system, which is the final output in this stage.
We use COCO dataset as our train and test dataset. The COCO dataset is a comprehensive collection used for object detection, segmentation, and captioning tasks. It is designed for scene understanding and is extracted from diverse and complex everyday scenes. Objects in images are precisely delineated through segmentation, enabling accurate positioning within the scene. The dataset comprises 91 object categories and includes 328,000 images with 2,500,000 labeled instances. As of now, it stands as the largest dataset for semantic segmentation, featuring 80 classes and over 330,000 images, with annotations available for 200,000 of them. The dataset encompasses more than 1.5 million object instances across various categories, providing rich and extensive data for object detection challenges. With an average of 7.2 objects per image, COCO poses a significant challenge for object detection tasks and remains one of the most widely used databases in the field.
We use NanoDet as our desired network. NanoDet is a lightweight object detection model designed for efficient inference on resource-constrained devices, such as embedded systems or edge devices. It is tailored to provide a balance between model size, speed, and accuracy, making it suitable for real-time applications where computational resources are limited. NanoDet typically utilizes streamlined architectures and optimization techniques to achieve its goals, often sacrificing some accuracy compared to larger and more complex models in exchange for faster inference times and lower resource requirements.

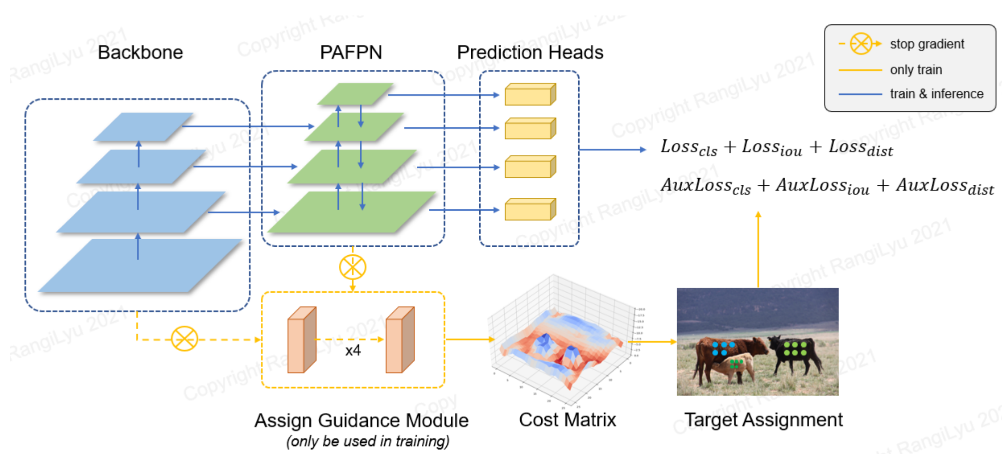

Figure 7: Visual model diagram

NanoDet indeed utilizes the Generalized Focal Loss as its loss function. By removing the centerness branch of FCOS, NanoDet reduces the computational overhead of the detection head, making it more suitable for lightweight deployment on mobile devices. This optimization helps in reducing the model's complexity while maintaining its effectiveness in object detection tasks, particularly in scenarios where computational resources are limited, such as mobile devices or embedded systems.

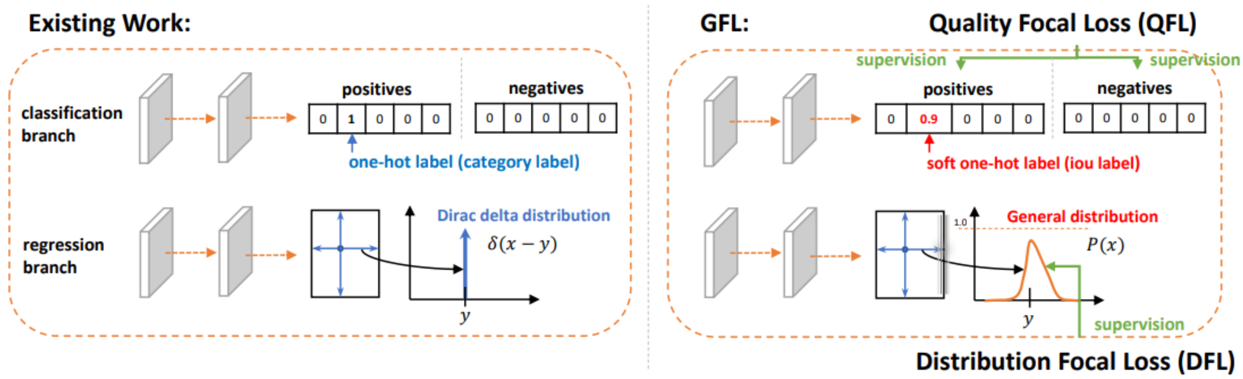

Figure 8: Visual model diagram

2.3.2 Requirements

1. Computational Resource Requirements:

- Inference Speed of NanoDet Model: Approximately 50 milliseconds to 200 milliseconds per image.
- Processor Performance: Utilize processors with moderate processing power, such as quad-core ARM Cortex-A53 or similar processors.

2. Memory Requirements:

- Memory Usage of NanoDet Model: Approximately 50MB to 200MB.

3. Camera and Image Processing Capability:

- Camera Resolution: $720p$ to $1080p$.
- Camera Frame Rate: At least 30 frames per second.
- Image Processing Capability: Ensure the system can perform real-time object detection at $1080p$ resolution.

4. Real-Time Requirements:

- Inference Latency: Targeting inference latency of no more than 200 milliseconds per image.
- Data Transmission Latency: Ensure that image data can be transmitted to the object detection model within 100 milliseconds, and detection results can be returned within the same time frame.

5. Environmental Adaptability:

- Adaptability to Lighting Conditions: Ensure that the object detection model can work stably under different lighting conditions.
- Adaptability to Background Complexity: Ensure that the object detection model can accurately detect objects in complex background environments.

2.4 Subsystem 3: Audio recognition

2.4.1 Overview

In the speech module, we first segment the speech signal through practical segmentation, dividing it into appropriate segments. Then, we use Convolutional Neural Networks (CNNs) to extract features from each segment, aiding in capturing crucial information within the speech signal. Subsequently, we input these features into a Bidirectional Long Short-Term Memory network (BiLSTM) for further processing. BiLSTM effectively models long-term dependencies in sequential data, thus better understanding contextual information within the speech signal. Ultimately, we use the output of BiLSTM as the target vector for subsequent speech recognition or other related tasks. This end-to-end process fully leverages the advantages of CNNs and BiLSTM, enabling efficient processing and accurate recognition of speech signals.

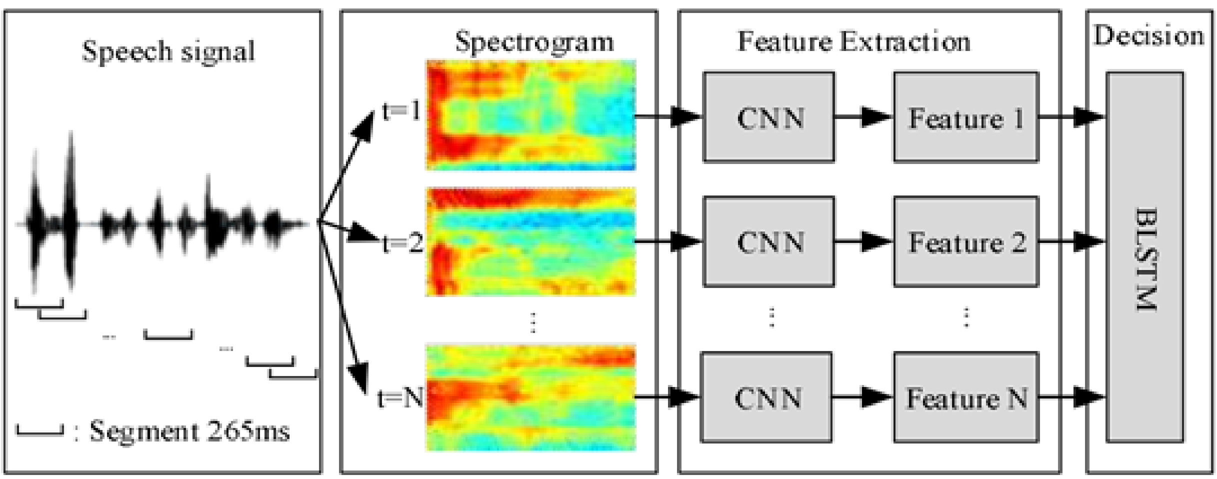

Figure 9: Audio diagram

2.4.2 Requirements

1. Memory Requirements:

- Memory Size: At least 100MB of memory space is required to load and run the models and data for the speech module.

2. Audio Capture Device:

- Microphone Performance: At least one basic microphone is needed to capture surrounding audio signals.

3. Real-Time Requirements:

- Inference Latency: Inference latency should be minimized, with the minimum acceptable latency for each speech segment not exceeding 200 milliseconds.
- Data Transmission Latency: Ensure that audio data can be transmitted to the speech module within 100 milliseconds, and processing results can be returned within the same timeframe.

4. Integration with Visual Module:

- Data Exchange Rate: Ensure that the data exchange rate between the speech module and the visual module is at least 1 Mbps.

5. Environmental Adaptability:

- Adaptability to Environmental Noise: The speech module should be able to operate stably in environments with low to moderate levels of environmental noise.

6. Safety and Reliability:

- System Stability: Ensure that the deployment of the speech module does not affect the stable operation of the robotic arm.

2.5 Tolerance Analysis

The first problem would be the imprecise movement of the robotics arm caused by the leak capability of the motor. A leaky motor can result in inconsistent movement patterns, causing the robotic arm to deviate from its intended path. This inaccuracy can lead to errors in positioning, impacting the arm's ability to perform tasks with precision. In tasks requiring delicate maneuvers or fine adjustments, such as in manufacturing or surgery, this imprecision could be critical. We would use control a PID controller to fix this problem.

The second problem would be the imprecise movement of the robotics arm caused by the leak capability of the motor. A leaky motor can result in inconsistent movement patterns, causing the robotic arm to deviate from its intended path. This inaccuracy can lead to errors in positioning, impacting the arm's ability to perform tasks with precision. In tasks requiring delicate maneuvers or fine adjustments, such as in manufacturing or surgery, this imprecision could be critical. We would use control a PID controller to fix this problem.

Another aspect is the error caused by exceeding the maximum weight the robotic arm can handle. While robotic arms are typically designed with a maximum load capacity in mind, uncertainties in the weight of objects being handled can lead to situations where the load exceeds this capacity. In such cases, the robotic arm may struggle to grip the object securely, resulting in errors in the gripping process. To address this issue, a load detection mechanism can be implemented in the control system to identify instances of overload and take appropriate measures, such as adjusting the gripping force or employing other stabilization techniques, to ensure the stability and accuracy of the gripping process.

Lastly, irregularities in the shape of the object being grasped can also contribute to gripping errors. Particularly with complex or irregularly shaped objects, the robotic arm may not be able to fully conform to the object's surface, leading to deviations in the gripping position. In such scenarios, optimizing the gripping strategy, such as using multi-point gripping or employing soft grippers, can improve the robotic arm's accuracy in handling irregular objects.

3 Ethics and Safety

Considering that our target group is mainly the elderly and disabled, ethics and safety are very important requirements in our design. In this section, we will divide into two parts to complete our proposal.

3.1 Ethics

In order to fulfill our obligations as ZJUI students and avoid ethical violations in the conduct and results of the project, our project team will strictly abide by the IEEE Code of Ethics [1] and ACM Code of Ethics [2]. We will undertake to fulfill but not limited to the following obligations and requirements:
1. Put the safety, health and welfare of the public first, strive to adhere to ethical design and sustainable practices, and have an obligation to report any signs of systemic risk that could lead to harm.
2. The project aims to contribute to society and human well-being by improving individual and team understanding of the capabilities and societal impact of traditional and emerging technologies.
3. Be honest and trustworthy, refrain from illegal acts in professional activities, and reject all forms of bribery.

3.2 Safety

In order to ensure the safety of the team and others, and to avoid any safety problems or hidden dangers during the project, our project team will strictly follow the ECE 445 SAFETY GUIDELINES [3]. We will undertake to fulfill but not limited to the following requirements:
1. No one on the team is allowed to work alone in the lab at any time.
2. In order to be allowed to work in the lab, everyone on the team must complete safety training.
3. Any group charging or using certain battery chemicals must read, understand, and follow safe battery usage guidelines.
4. The robot should include a switch that automatically stops in case of mechanical failure.

References

[1] IEEE. “”IEEE Code of Ethics”.” (2016), [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html (visited on 03/07/2024).
[2] ACM. “”ACM Code of Ethics”.” (2018), [Online]. Available: https://www.acm.org/code-of-ethics (visited on 03/07/2024).
[3] ZJUI. “”ECE 445 SAFETY GUIDELINES”.” (2024), [Online]. Available: https://courses.grainger.illinois.edu/ece445zjui/guidelines/safety.asp (visited on 03/07/2024).