Robomaster EP Core Voice Control

By Joseph Shotts on 6/19/23

Explanation: Using the robomaster EP Core is simple with the app on a tablet and thus not explained here. The Robomaster EP Core does have a Software Development kit that is designed to be simple and powerful. The SDK official docs can be found [here](https://robomaster-dev.readthedocs.io/en/latest/index.html) (https://robomaster-dev.readthedocs.io/en/latest/index.html). This documentation is about a python script that enables the robot to respond to a list of commands. **Usage of that voice control script and some parts of the SDK are explained here.** This project was designed for Windows machines and was tested on Windows 11. It was created by a beginner programmer (Joseph Shotts) and could be improved by more experienced developers.

Project Overview: The project was designed to use a lot of the functionality of the robomaster EP Core using voice commands. There is a list of the voice commands found later in this Doc. The robot can do things like dancing, simple movement, taking and dropping objects, and other functionality.

The python script uses a voice recognition model called Vosk because it is a simple voice recognition that works offline. The offline aspect is necessary because direct connection to the robot is much simpler than connecting through a router. This means that the computer running the script can't connect to the internet while it's connected to the robot. CMU Sphinx is another excellent and similar voice recognition option but it didn't want to work with Python 3.7.8 which is required for the robomaster SDK.

All that to say that the voice recognition definitely works but is not super high quality and frequently misses phrases. Future versions of this project would improve this capability. The speed of the program could also be improved. It uses multithreading where one thread reads an audio frame from the robot and passes it to the queue. The other thread reads this audio frame from the queue and converts it to text.

Using the voice control script: Using the voice control script is simple if python and the necessary libraries are installed.

1. Find python IDLE in applications and open it. It should open a shell.
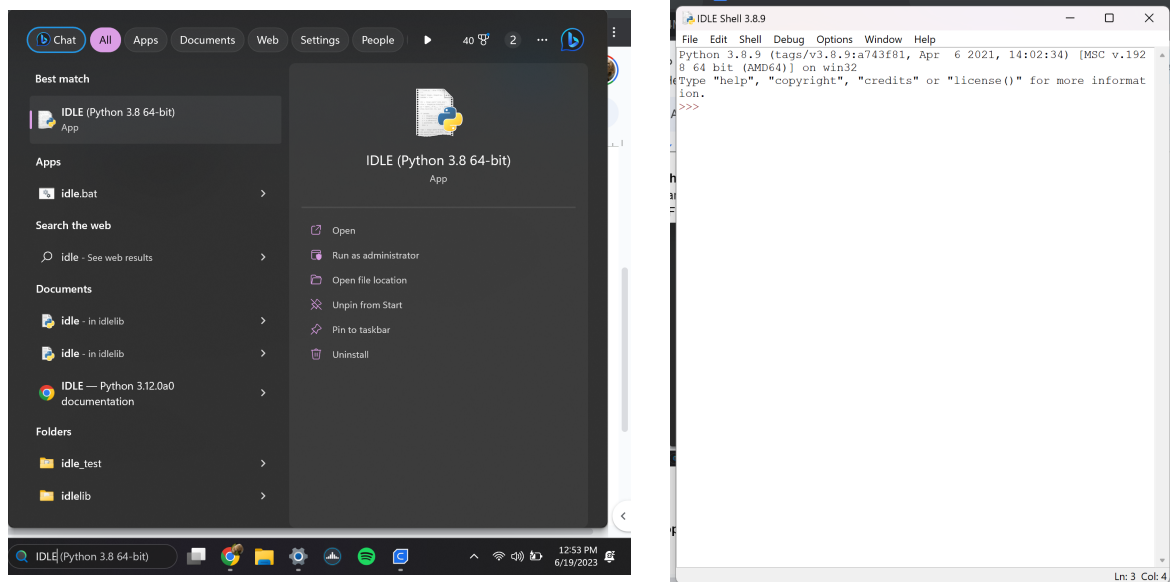

2. Select "File" -> "open" and then the file manager should open. The python script "Voice
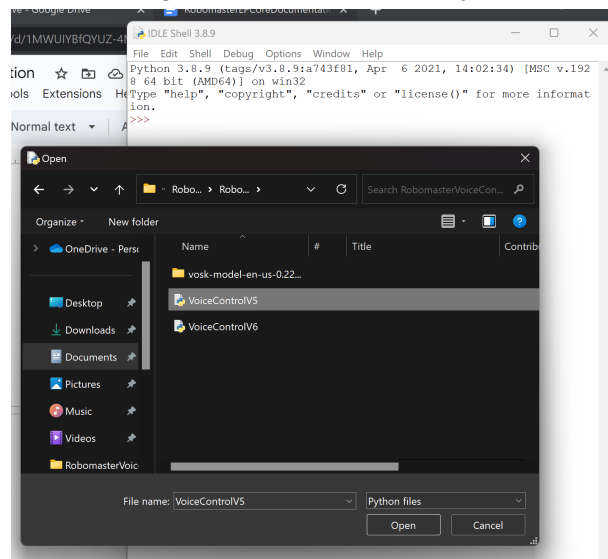
Control V8" needs to be opened.

3. The next step is to connect to the actual robot over wifi. The robot needs to be in direct connection mode and turned on. There is a switch on the robot as shown in the image below. If this switch is in the correct position simply connect to the wifi network labeled "RMEP-******". A password may be required. It should be "12341234" as that is the default for these robots.
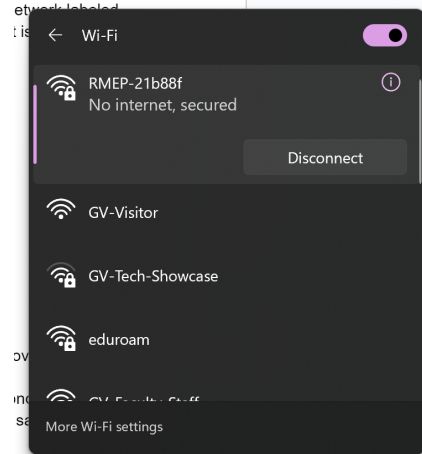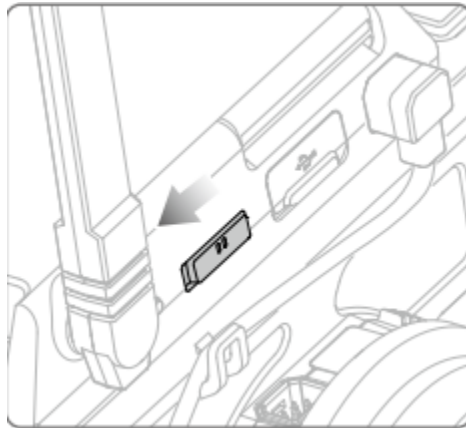
4. Finally select "Run" -> "Run Module". Assuming the robot is connected over wifi, the necessary libraries are installed, and the project folder (the folder "RoboamsterVoiceCOntrolV8") has all files intact the robot should respond to voice commands. You simply speak towards the robot's onboard camera and say "Jason, move forward" or any of his other commands starting with "Jason."
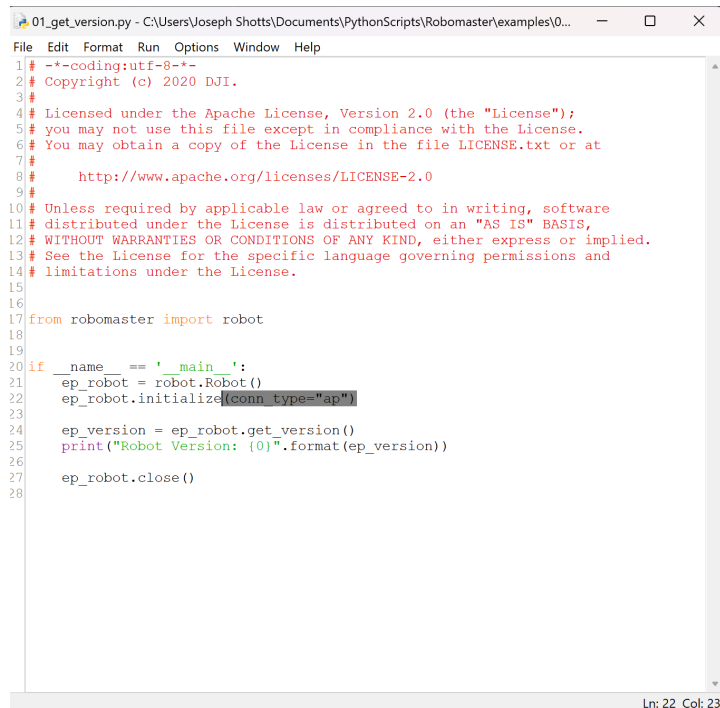

5. To stop the program simply select "shell" -> "Restart Shell" in the shell window.

List of Voice Commands:

FunctionName	What It Does	VoiceCommand
forward()	Move	forward'
backward()	Move	backward'
left()	Move	left'
right()	Move	turn'
turn_left()	left 45	turn left'
turn_right()	right 45	turn right'
flip()	rotate 180	flip'
spin()	rotate 360	spin'
donuts()	plays donuts audio file	donut'
do_the_bee_dance()	Like Wild Bees	do the be dance'
space_dance()	Dance	space dance
waltz()	Dance	walls
disco()	Dance	disco
jazz_hands()	Dance	jazz
rock()	Dance	rock
bluegrass()	Dance	bluegrass
follow_me()	Uses SDK voice recognition to try and follow someone for 15 seconds	
yes()	Moves Arm Up and Down	Yes/No Questions
no()	Moves Side To Side	Yes/No Questions
closeGrip()	close gripper	close'
openGrip()	open gripper	open', 'drop'
take()	Robot moves up opens grip, makes a sound, closes, and return to original position	take this'
give()	robot moves forward to nearest person, opens gripper	give'

Other optional scripts:

Included in this project drive are examples that may work on the robot as well. Simply connect to the robot over wifi and run the scripts in the terminal. Many of the scripts try to connect to the robot using a communication type called "sta" Make sure to switch to "ap" as shown below.

Setting up a computer to run the script:

1. Setup the python environment and install the robomaster SDK library. This can be done by following the tutorial found [here](https://robomaster-dev.readthedocs.io/en/latest/python_sdk/installs.html).
   (https://robomaster-dev.readthedocs.io/en/latest/python_sdk/installs.html) If this tutorial has been updated or appears incorrect screenshots of the original tutorial and necessary files can be found in the project drive in the folder titled "Python3.7.8AndRobomasterSDK"

2. Next Install Vosk by first running "pip install –upgrade pip" Then run "pip install Vosk".

3. Next Download the folder "RobomasterVoiceControlV5" from the project drive. It is important to make sure that the project folder is intact after downloading. If you open up the "RobomasterVoiceControlV5" Folder it should have a folder with Vosk voice recognition in the name as well as a bunch of .wav files and the script "VoiceRecognitionV5"

4. Put "RobomasterVoiceControlV5" in a folder that is easy to find. A good idea is to create a folder in documents called "PythonScripts." This folder is referred to in the "Using the voice control script" section. If you put it in a different spot you will open it from that spot.

5. IMPORTANT STEP: You have to download vosk-model-en-us-0.22-lgraph from https://alphacephei.com/vosk/models and add it to the project folder. Make sure that model folder is not nested inside another folder.