DEVEOPMENT OF AN AUTOMATED CHARGING SOLUTION FOR ELECTRIC VEHICLES

Bhagirathi bai V[1], Dr.Rahul Vivek Purohit[2], Dawn Mathew Mathew[3], Pratheek H Gadagi [4], Vedant V Mahadik[5], Sahil N Hon[6]

Article History: Received: 03.03.2023 **Revised:** 14.04.2023 **Accepted:** 09.06.2023

Abstract

The aim of the paper is to proposed automated EV charging bot is a cost-effective and easily installable solution for electric vehicle owners. It offers several standout features, including accurate homing capabilities, remote access control, and user prompting through telegram push notifications. The bot utilizes machine vision to locate the EV charging port and incorporates safety measures such as vehicle absence detection. It can also identify the vehicle's charge level using the manufacturer's API, promoting energy efficiency. By addressing the drawbacks of current charging solutions, the system aims to encourage wider adoption of electric vehicles and improve accessibility. Future enhancements may include battery swap technology for faster charging and increased range, integration with fleet management software, and the use of solar panels for sustainable power.

Keywords: Automated EV charging, Machine vision, Electric vehicle charging, Remote access, User prompting, Safety features, Charging port detection, Real-time overrides, Vehicle absence detection, Energy efficiency, Homing capabilities, Solar integration, Battery swap technology, Fleet management, Cost-effectiveness.

[1]Assistant professor, Department of Mechatronics engineering, Acharya Institute of Technology, Bangalore, India
[2]Assistant professor, B K Birla Institute of Engineering and technology, Pilani, Rajasthan, India
[3,4,5,6]Department of Mechatronics engineering, Acharya Institute of Technology, Bangalore, India

DOI: 10.31838/ecb/2023.12.S3.468

1. Introduction

In recent years, there has been a significant boom in the electric vehicle industry, with an increasing number of people opting for electric vehicles as a sustainable and environmentally friendly mode of transportation. With the advancement of technology, electric vehicles have become more affordable, efficient, and convenient, making them a viable alternative to traditional gasoline-powered vehicles. This boom in electric vehicles has been driven by several factors, including government initiatives, increased consumer awareness of environmental issues, and the emergence of innovative startups and established automobile manufacturers investing heavily in electric vehicle technology. As a result, the electric vehicle industry is experiencing a period of rapid growth and development, with new models, charging infrastructure, and battery technology emerging at an unprecedented rate. This boom in the electric vehicle industry is poised to transform the way we think about transportation and could play a significant role in reducing carbon emissions and mitigating the effects of climate change.

Many electric vehicles that we see are charged at the remote household garages. Manual charging of the electric vehicles leads to the most frequent issue faced by the electric vehicle owners, which is forgetting to charge the electric vehicle. This project addresses the issue where we have prototyped the automated electric charging station, where the user intervention is bare minimum, which is only for the consent to charge the electric vehicle.

We have designed a charging station hardware that would hold the charger and automatically plug the charger into the port once the car is parked in the vicinity and once user gives the permission to charge the vehicle.

Problem Definition

One of the challenges faced by electric vehicle owners is the issue of forgetting to charge their vehicles. This problem can lead to range anxiety, unexpected stops for charging, and delays in reaching their destinations. To address this challenge, several companies have developed smart charging solutions that can automatically charge electric vehicles. These solutions use advanced algorithms and communication protocols to enable the charging station to communicate with the vehicle and manage the charging process. Smart charging solutions can also be integrated with other smart home devices, such as voice assistants and mobile apps, to provide real-time updates on charging status and alerts if there is an issue with the charging process.

Objectives

Develop an automated EV charger by designing a low-cost, low-profile feedback-based solution.To Determine the Design considerations and the Structural Specifications for Mechanical frame.Calibration and optimization of the plug-in process.

Flowchart

Case 1: Automated charging with option for user prompt.
Figure 1 is the flowchart illustrating the complete flow of the process for the given case of user prompt.
Case 2: Completely automatic charging without user input.

Figure 2 is the flowchart illustrating the complete flow of the process for the given case without user prompt.

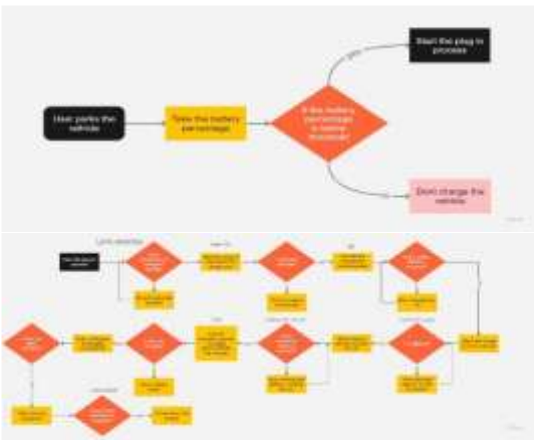

Fig 5.1 Case 1

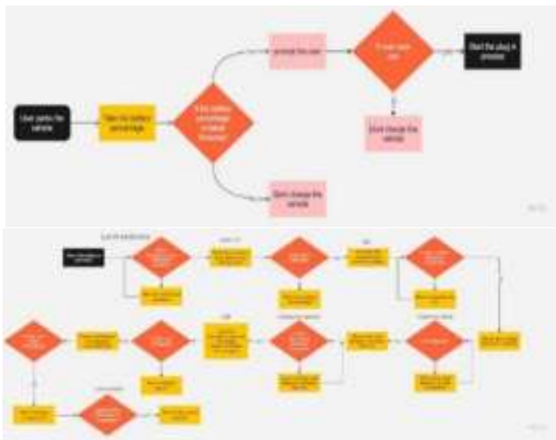

Fig 5.2 Case 2

Structural Framework

The parts involved in the complete assembly of automated EV charger are listed below with their design specifications and the mounting location in the assembly. Also, the significance of all the parts are specified with respect to the charging assembly.

C-plate: The C plate is typically placed on top of an aluminum extrusion, which serves as a guide rail for the movement of the structure. The C plate is designed with rollers on the inner side of the plate, which allows for smoother movement of the C plate in the X direction over the aluminum extrusion The rollers on the C plate are often made of materials that have low friction coefficients, such as polyethylene or nylon. These materials help to reduce the amount of friction and resistance encountered during movement, which in turn enables smoother and more efficient motion of the structure. The use of the C plate with rollers is a common design feature in many mechanical and industrial systems that require linear motion in the X direction. By facilitating smooth movement over the guide rail, the C plate helps to ensure that the structure moves precisely and accurately, which is often critical for the proper functioning of the overall system.

Fig 5.3 C plate 3

Guide rails: Guide rails are usually designed as a pair of parallel extruded aluminum rods that run along the length of the X axis. They are typically positioned at the top and bottom of the structure, and the C plate with rollers is placed between the two guide rails. The rollers on the C plate are designed to fit snugly against the guide rails, allowing them to slide smoothly and evenly along the length of the rails. This design helps to minimize friction and wear, ensuring that the motion of the structure remains consistent and reliable over time. Guide rails are a common feature in many mechanical and industrial systems that require linear motion in the X direction.

Front plate: The primary purpose of the front plate is to provide a platform for the charger and machine vision camera. The machine vision camera is also mounted on the front plate, and is used to help position the structure accurately with respect to the target charging port. The machine vision camera is often designed to capture images or video of the charging port, and uses software algorithms to calculate the distance, angle, and orientation of the structure relative to the target port. This information is then used to adjust the movement of the structure, allowing it to be positioned precisely for optimal charging. The front plate is often designed to be lightweight and durable, and may be made from materials such as aluminum or plastic. It is typically attached to the main structure using screws, bolts, or fasteners, and may be adjustable or re movable depending on the specific application.

Lead screw: The lead screw is a component that is used to facilitate the outward movement of the structure in the Z axis, allowing the charger to plug into the port that is located in front of the station. The lead screw is typically a long, threaded rod that is rotated by a motor or other power source. The thread on the lead screw is designed to mesh with a nut that is attached to the structure, allowing it to move up and down along the length of the screw as it rotates. In the assembly, the lead screw is attached to the center rod of the front plate. This allows the structure to be retracted and extended as needed, providing the necessary range of motion to reach the charging port.

Fig 5.4 Lead screw

Cylindrical rods: The steel rods are an integral part of the assembly that enables the movement of the aluminum extrusion in the X direction. These rods are cylindrical in shape and are placed in the provision provided on the C plate. The aluminum rods, attached to the cylindrical rods, move inward and outward due to their smooth sliding motion over the cylindrical rods. This movement is crucial for the overall functionality of the structure, as it enables the charger to be positioned accurately with respect to the target charging port. Moreover, the use of steel rods ensures the strength and stability of the assembly, making it robust and durable for extended use.

Timing belt: The timing belt is an essential component in the assembly of an automated EV charging mechanism. It functions as a transmission belt that runs over a motor and pulley at opposite ends of the top guide rail. The C plate is placed on the top guide rail and attached to the timing belt using a clamp. As the motor rotates, it pulls the timing belt, which in turn moves the C plate, allowing the smooth movement of the structure in the X-axis direction. This enables the plug-in process to occur without any hindrance, ensuring efficient and effective charging of electric vehicles. The timing belt is made up of a flexible material, usually rubber or polymer, with teeth along its inner surface that engage with matching teeth on the pulley. The teeth on the timing belt ensure that there is no slippage between the motor and the pulley, providing accurate and precise movement of the C plate.

Functionalities and Feature set

The developed automated Electric Vehicle (EV) charging bot is an innovative solution for charging electric vehicles, providing a hassle-free and autonomous charging experience. The developed system offers a range of features and functions that make them versatile and user-friendly. It development of the automated system was such that it can be deployed in various locations, such as parking lots, residential complexes, and public areas, to provide an efficient charging solution for EVs. The main use case and deployments are in residential applications as a part of home automation. Below will discuss the functions and features of automated EV charging bots, providing insights into their operation and how they can benefit EV drivers and the environment.

Basic functionalities:

The developed automated charging solution is a fully automated system that can detect, navigate, and charge EVs without human intervention. The basic functions of the automated charging bot include detecting the presence of an EV, automatically connecting the charging cable to the vehicle's charging port, and initiating the charging process. Once the charging process is complete, the bot will automatically disconnect the cable and return to its designated charging station. Some of the basic functions are:

Calibrated and controlled motion in two different axes: The high accuracy of the automated EV charging system is achieved through several mechanisms. Firstly, a closed-loop real-time position monitoring system is used to constantly monitor the machine's position and correct any errors in real-time. This compensates for any mechanical imperfections or external factors that may cause the machine to deviate from its intended path. Secondly, the motion planning algorithm is optimized to the machine's movements, taking into account factors such as acceleration, jerk, and feed rate, and is based on a trajectory generator that calculates the desired velocity profile for the machine's movements. Finally, the developed system uses high-resolution stepper motors that can move in small increments, allowing for fine control over the machine's position, and supports micro stepping, which further increases the resolution of the stepper motor and improves the accuracy of the machine's movements.

Fig 6.1 GRBL probe enable

Automated homing and workspace referencing: The developed system relies on homing for its proper working and work space referencing. The modified firmware is used to execute the homing process in a specific sequence, at a specific pace and within specific limits. The entire homing process was configured and calibrated for the developed mechanical structure with reliability and low cycle time as possible. Homing is a process that is used to establish the reference position of the bot. During the homing process, the machine is moved to a known position, using limit switches that are mounted on the machine. Once the machine reaches this position, it is considered to be "homed", and the coordinates of this position are used as the reference point for all subsequent operations. The homing process is essential for ensuring accurate and repeatable machining results, as it allows the machine to establish a known starting point for all operations. With respect to the application the bot was developed for, the bot should always be in the closed retracted position (homed) position before it searches for the EVs charging port, initiated the approach towards the vehicle for the plug-in process. By accurately homing the machine at the start of each work cycle, the machine is always starting from a consistent and repeatable position, which helps to minimize errors and improve overall accuracy. The following were configured for a proper homing process:

1. Homing Sequence
2. Homing seek speed
3. Homing debounce distance
4. Homing feed speed
5. Soft and Hard limits

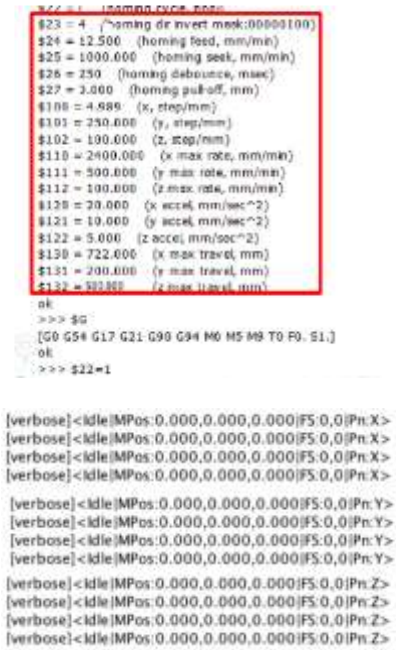

Fig 6.2 Verbose diagnostic check

Vehicle absence detection and Emergency Stop: The automated charging bot uses an image processing system to identify the EV charging port and initiate the charging process. It scans along the x-axis at a distance of 150mm, and if the port is not detected, it continues until it finds it. If it fails to find the port, it executes a homing process to return to the reference position and sends a message to the user indicating that the EV is not present. The soft limits are enabled within the work coordinates to trigger the homing sequence when the seeking sequence fails. The emergency stop (E-stop) feature is a safety mechanism that immediately stops all machine motion in case of an emergency. The E-stop is triggered by a physical switch that interrupts any ongoing operation and halts the bot's motion. The bot must be reset by restarting the control system to resume operation after the E-stop is activated.

Advanced Functionalities

The automated charging bot comes equipped with several advanced functions that enhance its overall performance and functionality. Some of these advanced features include the ability to perform real-time monitoring of the charging process, allowing the user to track the charging status and receive notifications when the charging is complete. The bot also incorporates a remote-control system, enabling the user to control the charging process from a remote location, making it convenient and user-friendly. Additionally, the bot can identify and avoid obstacles in its path, ensuring safe and efficient charging operation. Overall, the advanced functions of the bot provide added convenience, safety, and efficiency in the charging process, making it a valuable tool for EV owners. Some of the more advanced features are:

- Real Time Feedback
- User prompting and remote access
- Real time overrides
- Custom G codes

- Dynamic soft and hard limits

Real Time Feedback: Real-time feedback is an advanced feature that provides continuous feedback on the position of a machine's axes, allowing for precise control over movement. This is accomplished by using an encoder to track the movement of each axis and sending signals to the control algorithm to interpret and determine the position of each axis in real-time. The control system then compares the actual position of the machine to the desired position and adjusts or modifies the path if there is a discrepancy to ensure accurate and precise movement. The firmware uses a closed-loop feedback system, and the encoder used is a virtual one that sets a series of bits to zero once homed and changes the encoding bots depending on the movement of the machine. This feature is critical for ensuring accurate and efficient operation of the machine.

User prompting and remote access: In summary, user prompting and remote access are important features in the automated EV charging bot project, as they provide greater control and flexibility over the bot's functions. The runtime startup process ensures that the bot is always ready to function as soon as power is applied, while the Telegram chatbot allows users to remotely access and control the bot from anywhere with an internet connection. Status reporting is achieved through the Universal Gcode Sender (UGS) protocol, which provides real-time status updates on the bot's charging progress and position, and relays this information to the user via the chatbot. These features make the EV charging bot more efficient, user-friendly, and reliable.

Real time overrides: Real-time overrides are a set of advanced functions that allow for on-the-fly adjustments to the bot's behavior while it is in operation. In the context of this project, real-time overrides can be used to adjust the speeds of the bot while it is in the process of charging an EV. This can be useful in situations where the charging process needs to be slowed down or sped up due to changes in environmental conditions or other unforeseen circumstances. Additionally, real-time overrides can help to ensure that the charging process is as efficient as possible, by allowing for adjustments to be made in real-time to optimize the bot's behavior. These overrides can be initiated by the user via the Telegram chatbot, which sends the corresponding commands to the Raspberry Pi acting as the host machine. The Raspberry Pi, in turn, relays the command to the firmware, which executes the override in real-time. This way, the user can intervene if needed and make adjustments to the charging process in real-time, without needing to physically interact with the charging bot. Overall, real-time overrides are an important part of the project's functionality, as they allow for greater flexibility and control over the charging process.

Custom G codes: G codes are commands used to instruct the bot about different operations such as movement, specific speed, servo activation, and more. Custom G codes are user-defined codes that were added to create specific actions. These codes can be added to the firmware codebase and then compiled to create a custom version of the firmware. The benefit of custom G codes is that they can be tailored to meet specific needs and streamline the workflow. For example, custom G codes were created for the coordinated movement of X and Z movement at different speeds. Also a custom G Code involving the servo actuation by creating a PWM signal was also done.

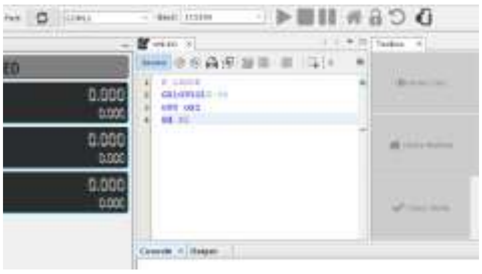

Fig 6.3 UGS custom G codes

Dynamic Soft and Hard limits: Dynamic soft and hard limits are advanced features that allow the user to set limits on the physical travel of the machine's axes in real-time. Unlike static limits, which are set once and remain constant, dynamic limits are flexible and can adapt to changes in the machine's position during operation. Dynamic hard limits are implemented through limit switches that trigger an emergency stop when the machine reaches its physical limits. These limits are set during operation, allowing for changes in the bot or EV positioning. Dynamic soft limits, on the other hand, provide a more advanced level of control over machine movement. Soft limits define the maximum and minimum allowable travel for each axis, and GRBL continuously monitors the machine's position to ensure it does not exceed these limits. If the machine approaches a soft limit, the firmware will slow down the feed rate to prevent overshooting the limit. This feature minimizes the risk of damage to the machine or workpiece due to accidental collisions. The soft limits are dynamic as well as they serve different functions depending on the operation being conducted.

Communication Framework

In the charging station system, communication is a crucial aspect in enabling remote control and monitoring of the system. Two types of communication are used: serial communication and server-based communication. Serial communication involves sending data bit by bit between two devices in close proximity, while server-based communication involves communication through a central server acting as a mediator between devices. Communication is used between the Raspberry Pi controller and the Telegram chatbot, enabling remote control of the system through the chatbot. The chatbot sends instructions to the Raspberry Pi controller, which executes actions on the hardware. Communication is also used between the GRBL and the Raspberry Pi controller, allowing the controller to send commands to the GRBL to control the stepper motors. These communication processes are essential for the charging station system to function effectively.

Rpi and telegram chatbot communication: Establishing communication between a Telegram chatbot and a Raspberry Pi is essential in building an automated system that can be remotely controlled and monitored. This communication can be established using the Telegram Bot API and the Python programming language. The communication process involves installing the required libraries, creating a Telegram bot and API token, setting up the Raspberry Pi, and writing the Python script to establish communication between the chatbot and the Raspberry Pi. In the case of the charging station system, the Telegram bot is used to control the charging process through the commands /start and /yes. The /start command initiates the charging process by plugging the charging station into the port and sending the current charge percentage of the car to the user. The system checks if the charge percentage is less than 60% before proceeding with further communication. The /yes command is then used to start the motors and initiate the charging cycle. After the charge is completed, the bot communicates with the user to inform them that the vehicle has been charged completely and the homing sequence has been initiated. In case the port is not found, the bot sends a message notifying the user.

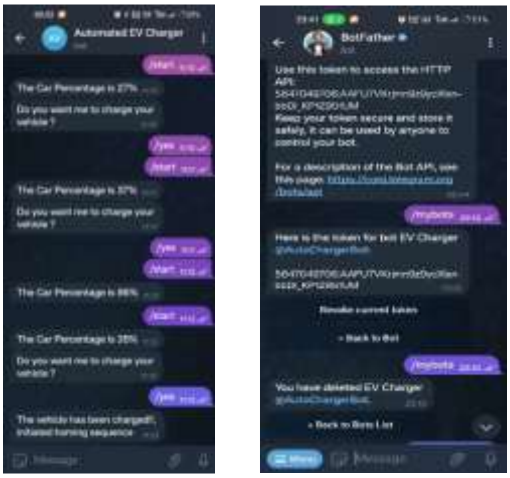

Fig 7.1 telegram Bot father and command interface

Rpi and GRBL communication: The Rpi and GRBL are connected serially for the purpose of communication. The significance of this communication is of immense importance in our system. The system mobility is controlled by the Stepper motor which controls the x and the z axis movement of the structure for the plugin process. The stepper motors are controlled by the motor drivers mounted on the CNC shield which receives the command serially via Rpi . The commands are written in the Python script as per the requirement. There are certain steps that needs to be followed in order to establish the communication:

Step 1: Connect the USB cable from the Rpi to the GRBL board.

Step 2: Install the PySerial library on the Rpi using the following command in the terminal:

```
sudo apt-get install python-serial
```

Step 3: Write a Python script to communicate with the GRBL. The script will use the PySerial library to establish the serial connection and send G-code commands to the GRBL board. The script has few aspects that makes the script different than any other python script.

```
serial.Serial()
```

Function is used to establish the serial connection with the GRBL board. The first argument is the device name (in this case, /dev/ttyACM0), and the second argument is the baud rate.

ser.write()
Function is used to send G-code commands to the GRBL board. The commands are sent as bytes using the b prefix. Once the communication is established, the Rpi can send G-code commands to the GRBL board to control CNC machines. The GRBL board will interpret the G-code commands and control the CNC machine accordingly. It is worth noting that the communication between the Rpi and GRBL is bi-directional, meaning that the GRBL board can also send messages back to the Rpi. This allows for real-time feedback and monitoring of the CNC machine's status.

Vitrtual Perception and Processing

Machine vision plays a crucial role in our automated EV charging bot project. By integrating cameras and image processing algorithms into our system, we can accurately and efficiently detect and locate the electric vehicle's charging port, which is essential for the charging process. Machine vision algorithms can analyze the images captured by the cameras and detect features such as the charging port's shape, size, and location. These algorithms can also correct for lighting and perspective distortions, ensuring that the charging port is accurately detected regardless of the lighting or angle of the camera. The use of machine vision in our automated EV charging bot project improves the system's overall accuracy and efficiency, enabling us to provide a seamless and convenient charging experience for electric vehicle owners.

Downloading and setting up open CV

OpenCV (Open Source Computer Vision) is a powerful open-source library for computer vision, machine learning, and image processing tasks. It provides a wide range of functions for image and video analysis, including image filtering, feature detection, object detection, tracking, and more. In our automated EV charging bot project, OpenCV is used to recognize and locate the EV charging port on the car, which is essential for the charging process. The software analyzes the images captured by the webcam and uses OpenCV algorithms to detect the position of the charging port. Once the port is located, the robot arm can be maneuvered to connect the charger to the port. Without OpenCV, this task would require manual intervention, and the charging process would not be fully automated. Therefore, the use of OpenCV in our project is crucial for achieving the automation and efficiency we desire.

Camera setup and initialization

On a Raspberry Pi running a Linux-based operating system, it requires additional steps to install the necessary drivers and libraries to access the webcam. Additionally, the Raspberry Pi has less processing power and memory than a typical laptop, so it may require more optimization of the OpenCV code to ensure efficient and accurate image processing. Finally, the configuration and setup of the webcam on a Raspberry Pi involves using command-line interfaces, text editors, and other tools that are not as commonly used in Windows environments.

Port detection and masking

Masking is a crucial technique in image processing, particularly in computer vision applications, such as our EV charging bot project. It involves selectively filtering out pixels from an image based on certain criteria to isolate specific features of interest. In our project, masking is used to detect the charging port of the electric vehicle. This is important because it allows our bot to accurately locate the charging port, align itself with it, and initiate the charging process. To implement the masking technique, we use HSV (hue, saturation, value) values to filter out all other pixels except those in the color range that corresponds to the charging port. This is because HSV is a more intuitive color space for human perception than RGB. To do this, we first convert the image from RGB to HSV color space, then define a range of HSV values that correspond to the color range of the charging port. We then create a binary mask by filtering out all pixels that do not fall within this color range. Finally, we apply this binary mask to the original image to extract only the charging port. This technique enables our EV charging bot to accurately detect and locate the charging port, even in different lighting conditions and with varying EV models. In order to obtain the HSV values a separate code was written to adjust and selectively chosen the object/ feature to identify. Making is done till on the required feature is seen in the positive od the mask. Once the mask is created, it is used to filter the image and extract the area of interest, which in our case is the EV charging port. The next step is to search for contours within the extracted area in real-time. A contour is a curve that connects the continuous points along the boundary of an object in an image. In our case, the contour represents the outline of the EV charging port. OpenCV provides functions to detect and draw contours on images. We use the "findContours" function in OpenCV to find the contours within the masked image. Once the contours are detected, we use the "approxPolyDP" function to approximate the contour with a polygonal curve to obtain the vertices of the EV charging port. These vertices are then used to calculate the center of the charging port, which is necessary to align the charging connector with the port. Parameters such as smallest and largest contour possible are also added to avoid false detection.

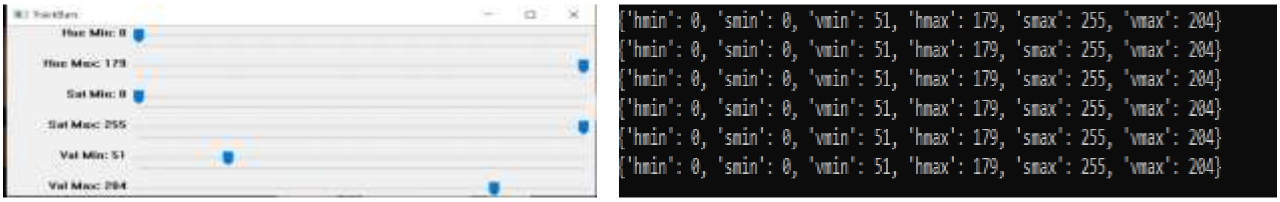

Fig 8.1 HSV values configuration

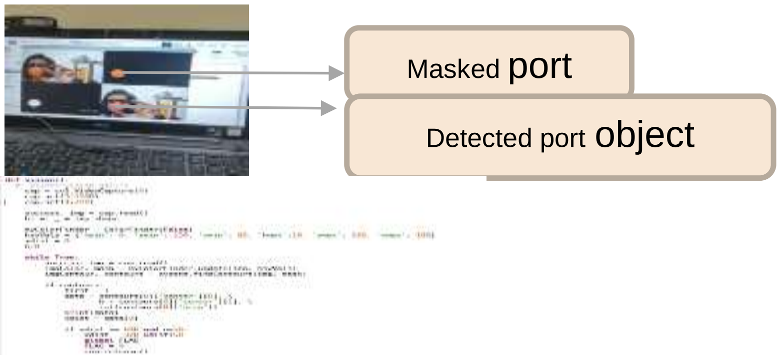

Fig 8.2 Port masking and detection

Coordinate Extraction:

Once the charging port contour is extracted from the image using the masking technique, the next step is to obtain its coordinates with respect to the camera frame. To achieve this, the contour is converted into a set of (x,y) coordinate pairs using the OpenCV findContours() function. These coordinates represent the location of the contour in the image plane. However, these coordinates are in pixel units and do not directly relate to the physical position of the charging port. To obtain the physical position of the charging port, the pixel coordinates need to be transformed using camera calibration parameters. These parameters include the intrinsic camera matrix, distortion coefficients, and extrinsic camera parameters such as the rotation and translation vectors. These parameters can be obtained using a camera calibration process, where a set of known calibration images are used to estimate the camera parameters. Once the camera parameters are obtained, they can be used to transform the pixel coordinates into real-world coordinates, typically in millimeters.

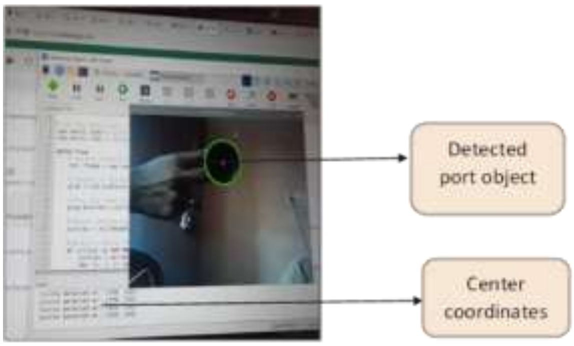

Fig 8.3 Port Coordinate Extraction

Positioning and Alignment of the Bot

The coordinates obtained from the contour detection need to be transformed from the camera frame to the bot's coordinate frame before they can be used to move the bot. This transformation involves several steps. First, the camera's intrinsic parameters such as focal length, principal point, and distortion coefficients need to be determined. This can be done using camera calibration techniques. Once the intrinsic parameters are known, the extrinsic parameters which describe the position and orientation of the camera with respect to the bot's coordinate frame can be determined. Next, the pixel coordinates of the detected contour in the camera frame are converted to normalized image coordinates by dividing them by the width and height of the image. The normalized image coordinates are then undistorted using the distortion coefficients obtained from camera calibration. After this, the

3D coordinates of the contour in the camera frame are obtained using the inverse perspective mapping technique. This involves computing the depth of the contour using its apparent size in the image and the known physical dimensions of the charging port. Finally, the 3D coordinates of the contour in the camera frame are transformed to the bot's coordinate frame using the extrinsic parameters obtained earlier. The resulting coordinates can then be used to move the bot to the correct position for charging.

Fig 8.4 Code block for Plug alignment

2. RESULTS

The automated electric vehicle charging system prototype that was developed successfully achieves its intended purpose of providing a reliable and efficient way of charging an electric vehicle. The system is able to detect the presence of the charging port of the electric vehicle using computer vision techniques and then align the charging plug to the charging port with a high degree of precision. The system makes use of various hardware components such as the stepper motors, servo motors, and the CNC shield, which are controlled by the software developed using the Arduino IDE and Python Open CV. The software processes the video feed from the webcam, masks the unwanted portions of the image, and detects the positively masked charging port. It then sends G codes to the CNC shield, which in turn controls the stepper motors for precise movement of the charging plug. The prototype is capable of performing movements in the X and Z directions, with a total extension of up to 0.3 meters and a total range of movement in the X direction of 0.722 meters. The system also has a stored length, which gives the distance from the wall to the end of the structure when it is in the home position. Overall, the developed prototype provides a reliable, efficient, and cost-effective solution for electric vehicle charging, and can be a valuable addition to parking lots, garages, and other public charging facilities. The operation of the automated charging solution for automotive vehicles developed can be divided into several stages. Each stage plays a crucial role in ensuring the smooth and efficient functioning of the system. There are 2 cases in which the operation sequence can be followed by the system.

Case 1: When the vehicle is in the charging station.

Homing sequence: During the homing sequence, the system follows a predefined set of steps to locate and establish its reference position. The homing sequence ensures that the automated system has a reliable and consistent starting point, allowing for accurate and repeatable operations. It establishes a foundation for subsequent actions, contributing to the system’s overall precision and reliability.

Vehicle Detection: The first stage involves the detection of the automotive vehicle that requires charging. This can be achieved through camera as a detection mechanism. The system is designed to identify the presence and position of the vehicle accurately.

Alignment and Connection: Once the vehicle is detected, the system proceeds to align the charging mechanism with the charging port of the vehicle. This stage involves precise mechanical movements to establish a proper connection between the charging station and the vehicle’s charging port. The system ensures that the alignment and connection are secure and reliable.

Fig 9.1 Alignment and connection

Charging Completion and Disconnection: Once the vehicle’s battery reaches the desired charging level or completes the charging cycle, the system initiates the disconnection process. It ensures a safe disconnection of the charging mechanism from the vehicle’s charging port. The system provides notification or indications to inform the user that the charging process is complete. System Reset and Ready State: After the disconnection, the automated charging solution resets itself and prepares for the next vehicle charging. It goes back to its ready state, awaiting the detection of another vehicle that requires charging.

Case 2: When the vehicle is not present in the station.

When the automated charging system does not detect a car, it follows a specific set of operations to handle this situation. Here is a brief explanation of the system’s actions when a car is not detected:
Homing sequence: During the homing sequence, the system follows a predefined set of steps to locate and establish its reference position. The homing sequence ensures that the automated system has a reliable and consistent starting point, allowing for accurate and repeatable operations. It establishes a foundation for subsequent actions, contributing to the system’s overall precision and reliability.
X-direction Movement: The system moves in the X-direction for a predefined distance. This movement can be achieved using motors, actuators, or any other mechanism capable of controlled motion.
Charging Port Detection: After moving in the X-direction, the system searches for the charging port of the car. This detection process may involve sensors or cameras specifically designed to identify the charging port.
Port Not Detected: In case the charging port is not detected after the X-direction movement, the system repeats step 2 and 3. It moves the same predefined distance in the X-direction again, scanning for the charging port Iterative Movement: The system continues to move in the X-direction and perform charging port detection until it reaches the extreme position in the X-direction. This ensures that the entire range is covered in the search for the charging port.

Fig9.2 Complete seeking

Car Not Detected: If the system does not detect the car’s charging port at any point during the iterative X-direction movement, it determines that the car is not present in the charging station. User Notification: The system sends a Telegram message or notification to the user, indicating that the car was not detected. This informs the user about the status of the charging process and prompts appropriate action if necessary.

Fig 9.3Telegram chatbot Communication for no car detected

Homing Sequence Initiation: After notifying the user, the system starts the homing sequence. This sequence allows the system to return to a known reference position or home position to ensure accuracy and repeatability for future operations.

3. REFERENCE

1. Automatic vehicle recharging station (AVRS) by WPI on 25th april, 2019 by, Matthew Fortmeyer,Nikolas Gamarra,Robert O’Brien,Jacob Remz.
2. Design of automatic plug in system based on machine vision by, You Bo, Wu kun, Xu Jiazhong, Chen Guojie,Liang Qiang, Tian Jiande.
3. 3D vision guided robotic charging station for electric and plug in hybrid vehicles by Cornell university on 15 th march 2017 by, Justinas Miseikis,Matthias Ruther, Bernhard Walzel, Mario hirz,Helmut Brunner
4. Automated driving ,2018 by Acosta, L., Blanco, R., Haber, R., Villagra, J., Naranjo, J., Navarro, P., Sánchez, F.
5. Davis, M. R. (2018, Mar 26). Electric cars vs. gas cars: Comparing maintenance & battery costs. Ez-Ev. [6] “EV-Volumes - The Electric Vehicle World Sales Database,” http://www.ev-volumes.com/country/total-world-plug-in-vehicle-volumes/, (Accessed on 03/08/2017).

6. “HALCON The power of machine vision – MVTecSoftwareGmbH,”http://www.mvtec.com/products/halcon/
7. M. Silverman, B. Jung, D. Nies, and G. Sukhatme, “Staying alive longer: Autonomous robot recharging put to the test,”Center for Robotics and Embedded Systems (CRES) Technical Report CRES, vol. 3, p. 015, 2003.
8. Dilip R, Bhagirathi V. (2013) Image Processing Techniques for Coin Classification Using LabVIEW. OJAI 2013, 1(1): 13-17 Open Journal of Artificial Intelligence DOI:10.12966/ojai.08.03.2013 12.
9. Naveen Mukati, Neha Namdev, R. Dilip, N. Hemalatha, Viney Dhiman, Bharti Sahu, Healthcare Assistance to COVID-19 Patient using Internet of Things (IoT) Enabled Technologies, Materials Today: Proceedings,2021,https://doi.org/10.1016/j.matpr.2021.07.379. ISSN214-7853,
10. Mr. DILIP R, Dr. Ramesh K. B. (2020). Development of Graphical System for Patient Monitoring using Cloud Computing. International Journal of Advanced Science and Technology, 29(12s), 2353 - 2368.
11. Mr. Dilip R, Dr. Ramesh K B ."Design and Development of Silent Speech Recognition System for Monitoring of Devices", Volume 7, Issue VI, International Journal for Research in Applied Science and Engineering Technology (IJRASET) Page No: , ISSN : 2321-9653
12. R. Dilip, Y. D. Borole, S. Sumalatha and H. Nethravathi, "Speech Based Biomedical Devices Monitoring Using LabVIEW," 2021 9th International Conference on Cyber and IT Service Management (CITSM), 2021, pp. 1-7, doi: 10.1109/CITSM52892.2021.9588853.
13. Pandey, J.K. et al. (2023). Investigating Role of IoT in the Development of Smart Application for Security Enhancement. In: Sindhwani, N., Anand, R., Niranjana Murthy, M., ChanderVerma, D., Valentina, E.B. (eds) IoT Based Smart Applications. EAI/Springer Innovations in Communication and Computing. Springer, Cham. https://doi.org/10.1007/978- 3-031-04524-0_13.
14. Dilip, R., Samanvita, N., Pramodhini, R., Vidhya, S.G., Telkar, B.S. (2022). Performance Analysis of Machine Learning Algorithms in Intrusion Detection and Classification. In: Balas, V.E., Sinha, G.R., Agarwal, B., Sharma, T.K., Dadheech, P., Mahrishi, M. (eds) Emerging Technologies in Computer Engineering: Cognitive Computing and Intelligent IoT. ICETCE 2022. Communications in Computer and Information Science, vol 1591. Springer, Cham. https://doi.org/10.1007/978-3-031-07012-9_25.
15. Anurag Shrivastava, Chinmaya Kumar Nayak, R. Dilip, Soumya Ranjan Samal, Sandeep Rout, Shaikh Mohd Ashfaque, automatic robotic system design and development for vertical hydroponic farming using IoT and big data analysis,Materials Today: Proceedings,2021,ISSN 2214-7853, https://doi.org/10.1016/j.matpr.2021.07.294.
16. Gupta, N. ., Janani, S. ., R, D. ., Hosur, R. ., Chaturvedi, A. ., & Gupta, A. . (2022). Wearable Sensors for Evaluation Over Smart Home Using Sequential Minimization Optimization-based Random Forest. International Journal of Communication Networks and Information Security (IJCNIS), 14(2), 179–188. https://doi.org/10.17762/ijcnis.v14i2.5499
17. Pratik Gite, Anurag Shrivastava, K. Murali Krishna, G.H. Kusumadevi, R. Dilip, Ravindra Manohar Potdar, under water motion tracking and monitoring using wireless sensor network and Machine learning,Materials Today: Proceedings,2021,ISSN 2214- 7853, https://doi.org/10.1016/j.matpr.2021.07.283.
18. Deepthi, B. L., and Deepti Naik. "A Short Review of Industrial Pollution Prediction and Controlling Systems." Journal of Remote Sensing GIS & Technology (2023): 41-46.
19. Solabagoudar, Manjunatha P., et al. "A Review of Surveillance and Fire Fighter Drone." Journal of Mechanical Robotics (e-ISSN: 2582-2187) 8.2 (2023): 1-7.