CONSTRUCTOR UNIVERSITY

Duckietown Project :
Ball tracking and gesture detection

By Group 4: Saad El Jadouri
Yadav
Haseeb Ahmed

Spring 2023:

Abstract:

Using the Duckietown Platform, This project aims to include the ideas of Ball Tracking and Gesture Recognition. Autonomous driving and artificial intelligence are combined in the open sources robotics platform Duckietown. Duckietown's capabilities will be increased by adding Ball tracking and Gesture recognition, which will increase its adaptability and range of potential uses. Moreover, the project sheds light on the functionality, precision, and prospective uses of the improved Duckietown system through experimental assessments and analysis. By advancing robotics and AI research, this effort makes it possible to create more interactive and intelligent autonomous systems.

1.1) Introduction:

Autonomous driving and artificial intelligence are combined in the open-source robotics platform Duckietown. The "Duckiebots" are little, autonomous cars that operate in a simulated urban setting, navigating streets, adhering to traffic laws, and interacting with their surroundings. Duckietown's main goal is to offer a practical learning environment where researchers, students, and teachers may learn about robotics, autonomy, and artificial intelligence.

The platform can now do new things thanks to the addition of ball tracking and gesture recognition features in Duckietown. Duckiebots can perform activities like object manipulation, target following, or even interactive games thanks to ball tracking, which enables them to find and follow a ball in the surroundings. With this feature, Duckietown becomes more adaptable and user-friendly while also increasing user engagement and interaction.

On the other hand, Duckiebots can understand and comprehend human motions thanks to gesture recognition. Users may communicate with the Duckiebots through simple hand gestures thanks to this integration, which makes it easier for humans and robots to engage. Duckiebots can conduct actions or reply to human requests by comprehending and responding to gestures, boosting the user experience and facilitating natural interaction.

Motivation/Objectives:

This initiative has two objectives. First and foremost, we want to use reliable ball tracking algorithms in the Duckietown setting. This entails creating algorithms that can detect and track a ball precisely even in dynamic and difficult situations. Yadavbot can successfully interact with the ball and carry out a variety of tasks based on its position and movement by obtaining accurate ball tracking.

The second goal is to integrate gesture recognition features into Duckietown. Creating machine learning models or algorithms that can decipher human gestures and categorize them into useful commands or actions is required for this. Ourbot can respond to user inputs by recognizing gestures, enabling intuitive and natural interaction between humans and robots.

Duckietown's instructional value will be increased, and its application possibilities will be increased, by incorporating ball tracking and gesture recognition. In order to explore the fields of ball tracking, gesture recognition, and human-robot interaction, the project aims to offer researchers and students a thorough framework. In the end, our work promotes creativity and enables the creation of more interactive and intelligent robots in a variety of real-world settings, furthering the growth of robotics, AI, and autonomous systems.

1.2) Problem formulation:

We plan to create powerful item detection and tracking features for the Duckietown platform. Robots are better equipped to perceive and interact with their surroundings when object detection and tracking are performed as essential computer vision tasks. Duckiebots will be able to recognize and track objects in their environment, such as balls, obstacles, or other important entities, if these capabilities are implemented into Duckietown.

Object Detection:

Identifying and localizing particular items inside a still or moving picture or video frame is known as object detection. The goal in Duckietown is to use cameras or onboard sensors put on the Duckiebots to find objects of interest, such as balls. The method for detecting objects is described in the following steps:

a. Acquisition of Sensor Data: Using their built-in cameras or sensors, the Duckiebots gather sensor data, such as pictures or video feeds.

b. Preprocessing: The obtained sensor data is enhanced and any noise or distortions are eliminated during preprocessing. Image scaling, normalization, and noise reduction processes could be used at this step.

d. Object Localization: In an image or video frame, detected objects are often represented by bounding boxes that show how far they extend spatially. For further tracking, these bounding boxes offer the position and size details required.

Object/Ball Tracking: Object tracking entails following and updating a detected object's position throughout time continuously. By using tracking, Duckiebots can keep track of an object's trajectory, speed, and direction, ensuring that it remains in the robot's field of view. The following phases make up the object tracking process:

a.Initialization: The tracking algorithm initializes a tracker and correlates it with the detected object after an object is discovered. In order to provide a reference for later tracking, initialization entails extracting features or descriptors from the initial bounding box.

b. Motion Estimation: By comparing the object's features or descriptors to the original reference, the tracker analyzes successive frames and guesses the object's motion. The tracker can forecast the object's position in upcoming frames thanks to this estimation.

c. State Update: The tracker adjusts the object's projected position as new frames come in, taking into consideration the object's anticipated motion. Based on the observed motion and any changes in appearance, the bounding box coordinates and size of the item are modified during this process.

d. Occlusion Handling: Mechanisms for handling occlusion, in which the tracked item may be momentarily hidden by other objects or obstacles, are frequently included in object tracking algorithms. When an object reappears, these processes use methods like motion prediction, feature matching, or re-detection to determine its position.

Yadavbot can observe and engage with their surroundings more successfully by incorporating object detection and tracking within Duckietown. By combining these qualities, the platform becomes more autonomous and provides opportunities for a variety of tasks, including cooperative tasks, ball manipulation, and obstacle avoidance. Furthermore, the creation and improvement of object recognition and tracking algorithms promote innovation and allow for the deployment of more sophisticated and effective autonomous systems while also advancing the fields of computer vision and robotics.

Gesture Recognition:

In addition, we want to use gesture controls to operate the Duckiebot as part of this project. Users can engage naturally and intuitively with the Duckiebot by employing basic hand movements thanks to the incorporation of gesture recognition. The implementation of gesture recognition and the Duckiebot's accompanying behavior are described in more detail below:
We have used Handlandmarks Mediapipe Algorithm for Gesture Recognition:

Handlandmarks mediapipe: MediaPipe is an open-source framework developed by Google that provides a wide range of machine learning solutions for various computer vision tasks. One of its notable features is the hand tracking module, which enables hand landmark detection and tracking.

The trained model is put to use on the Duckiebot to recognize gestures in real-time. It analyzes the previously processed frames and forecasts the associated gesture label.
When a gesture is identified, it understands and carries out the corresponding behavior.

Gesture-Based Behaviour:

The bot executes programmed behaviors or activities in accordance with the recognized gesture.
As an illustration, Yadavbot will halt its movement and remain motionless if the user makes the "stop" gesture.
A "forward" gesture from the user causes the bot to begin traveling at a set speed.
Similar associations can be made between other gestures like "left," "right," or "backward" and the corresponding behaviors.
The steering angle and speed of the Duckiebot are adjusted as necessary by the control system, which translates the recognized gesture into the proper movement orders.

We give the Duckiebot the ability to recognize and interact with the ball in the Duckietown environment by implementing object identification and tracking algorithms, particularly by using Hough transformations for ball detection. The combination of these abilities improves the Duckiebot's autonomy and opens up possibilities for activities like manipulating balls, following targets, and playing interactive games.

The efficient use of the tracking algorithm, movement control, and ball recognition using Hough transforms exemplifies the flexibility and possible uses of the Duckietown platform.

The Duckiebot can now identify and follow a ball on its own, interacting with its surroundings and reacting to the ball's movements.
Overall, the Duckietown platform's incorporation of gesture detection and behavior management marks a significant advancement toward the creation of interactive, user-centered robotics systems. This project promotes more investigation and study in the fields of gesture detection, human-robot interaction, and control mechanisms, laying the foundation for later developments in user-friendly robot control interfaces.

Overall, the Duckietown platform's incorporation of gesture detection and behavior management marks a significant advancement towards the creation of interactive, user-centered robotics systems. This project promotes more investigation and study in the fields of gesture detection, human-robot interaction, and control mechanisms, laying the foundation for later developments in user-friendly robot control interfaces.

of the ball.Then if in case our bot identified the ball it starts moving towards it.

2.1) Computer Vision methods:

Hough:The purpose of the technique is to find imperfect instances of objects within a certain class of shapes by a voting procedure.We used it to identify the ball.

Canny:It treats edge detection as a signal processing problem. The key idea is that if you observe the change in intensity on each pixel in an image, it's very high on the edges. In this simple image below, the intensity change only happens on the boundaries.In our project it is used to calculate the center of the ball.

Opencv:is a Python library that allows you to perform image processing and computer vision tasks. It provides a wide range of features, including object detection, face recognition, and tracking.We used it for Hough and Canny as well as to perform other Computer Vision methods.

2.2) Architecture :

DTS:Duckietown Shell is a pure Python, easily distributable (few dependencies) utility for Duckietown.
The idea is that most of the functionality is implemented as Docker containers, and dt-shell provides a nice interface for that, so that user should not type a very long docker run command line.

Ros:The Robot Operating System (ROS) is a set of software libraries and tools that help you build robot applications. From drivers to state-of-the-art algorithms, and with powerful developer tools.

Docker:Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers.