Autonomous Navigation of Mars Rover

Abhishek Nikhil Garima

Objective and Overview

The objective of this project is to develop a system which helps the rover to navigate and move autonomously. We will explore various algorithms and learn computer vision techniques to accomplish our task. We will basically be providing eyes and brain to the rover.

- Object detection and classification using image processing and deep learning
- Depth estimation of the objects by using the 3D Zed Camera
- Localization of the rover by map creation for the rover by using SLAM

Object Detection

Convolution is the process in which each element of the image is added to its local neighbors, and then it is weighted by the kernel.A convolution kernel is a 2D structure whose coefficients define the characteristics of the convolution filter that it represents. In a typical filtering operation, the coefficients of the convolution kernel determine the filtered value of each pixel in the image.

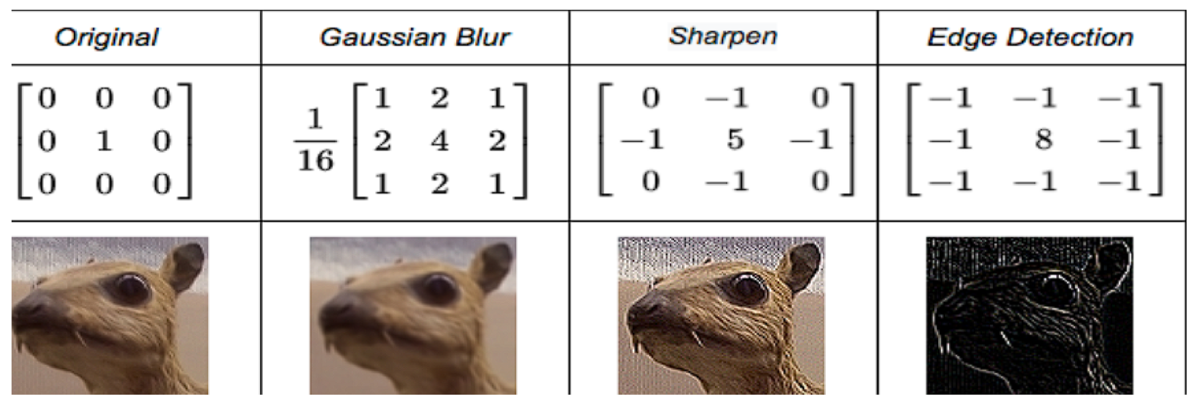

Figure 1. Operations on a Kernel

Neural networks can help computers make intelligent decisions with limited human assistance. This is because they can learn and model the relationships between input and output data that are nonlinear and complex.The neural networks consist of interconnected nodes or neurons that process and learn from data, enabling tasks such as pattern recognition and decision making in machine learning.

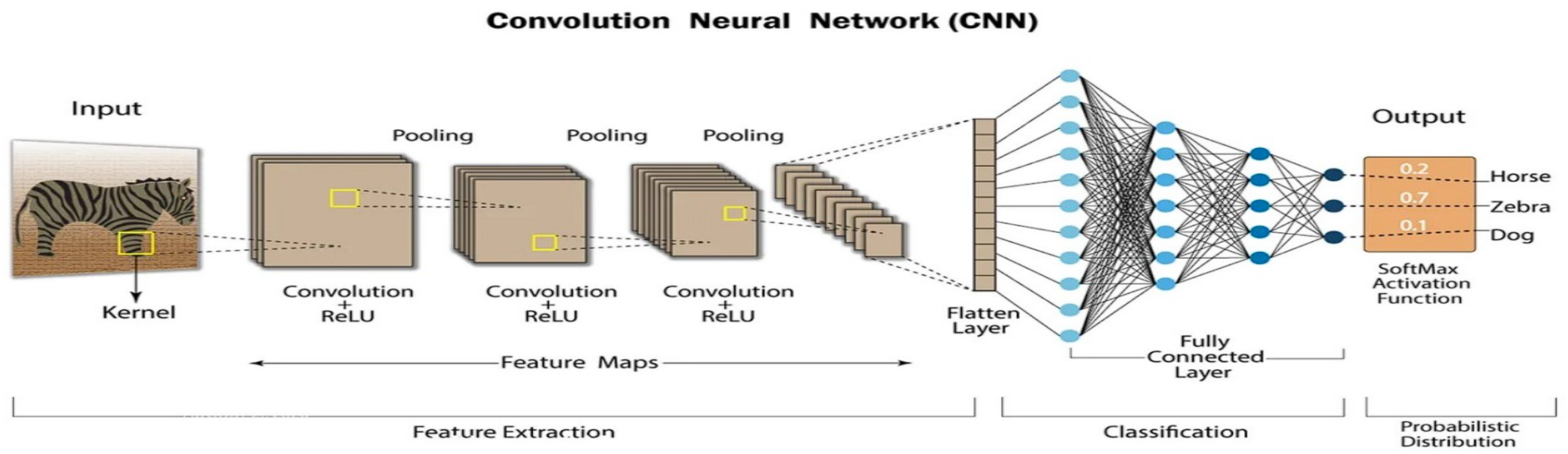

Figure 2. Convolutional Neural Network

Some algorithms used for the Object Detection:

- **Canny Edge Detection** It begins with noise reduction using a Gaussian filter to smooth the image and reduce unwanted details. Next, the intensity gradients of the image are calculated, often using Sobel operators. The algorithm then applies non-maximum suppression to thin out the edges. Following this, double thresholding is used to classify edges as strong, weak, or non-relevant based on two threshold values, ensuring that strong edges are definitely part of the edge map. Finally, edge tracking by hysteresis is performed to preserve weak edges connected to strong ones and discard the rest.
- **Hough Transform** The Hough transform is a feature extraction technique used in digital image processing. The purpose of the technique is to find imperfect instances of objects.

YOLOv8: An Object Detection Model

You Only Look Once (YOLO) proposes using an end-to-end neural network that makes predictions of bounding boxes and class probabilities all at once. It differs from the approach taken by previous object detection algorithms, which repurposed classifiers to perform detection.

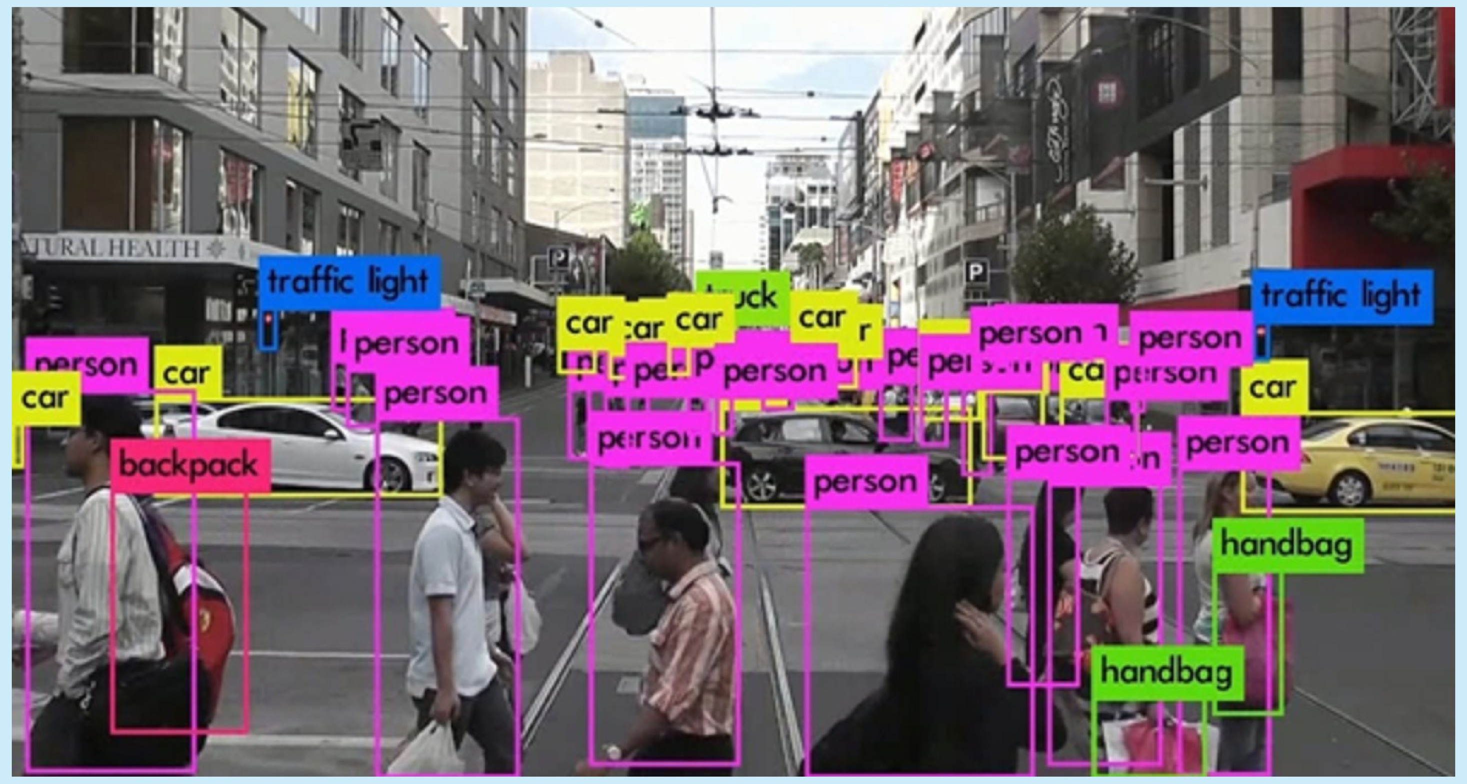

Figure 3. Image trained on YOLOv8

Predicting Relative Depth

The pre-trained YOLOv8 model is loaded to process the image and detect objects with a low confidence threshold, ensuring all potential rocks are detected, and then outputs the bounding boxes for detected objects.
The depth information for the image is used to extract the average depth value within each bounding box, representing the typical distance of the detected object from the camera.
To determine the relative position of the object, the average depth is normalized against the entire depth range in the scene.
The object's position is categorized as "FAR AWAY" or "CLOSE" based on predefined thresholds, providing a qualitative understanding of its proximity to the camera.

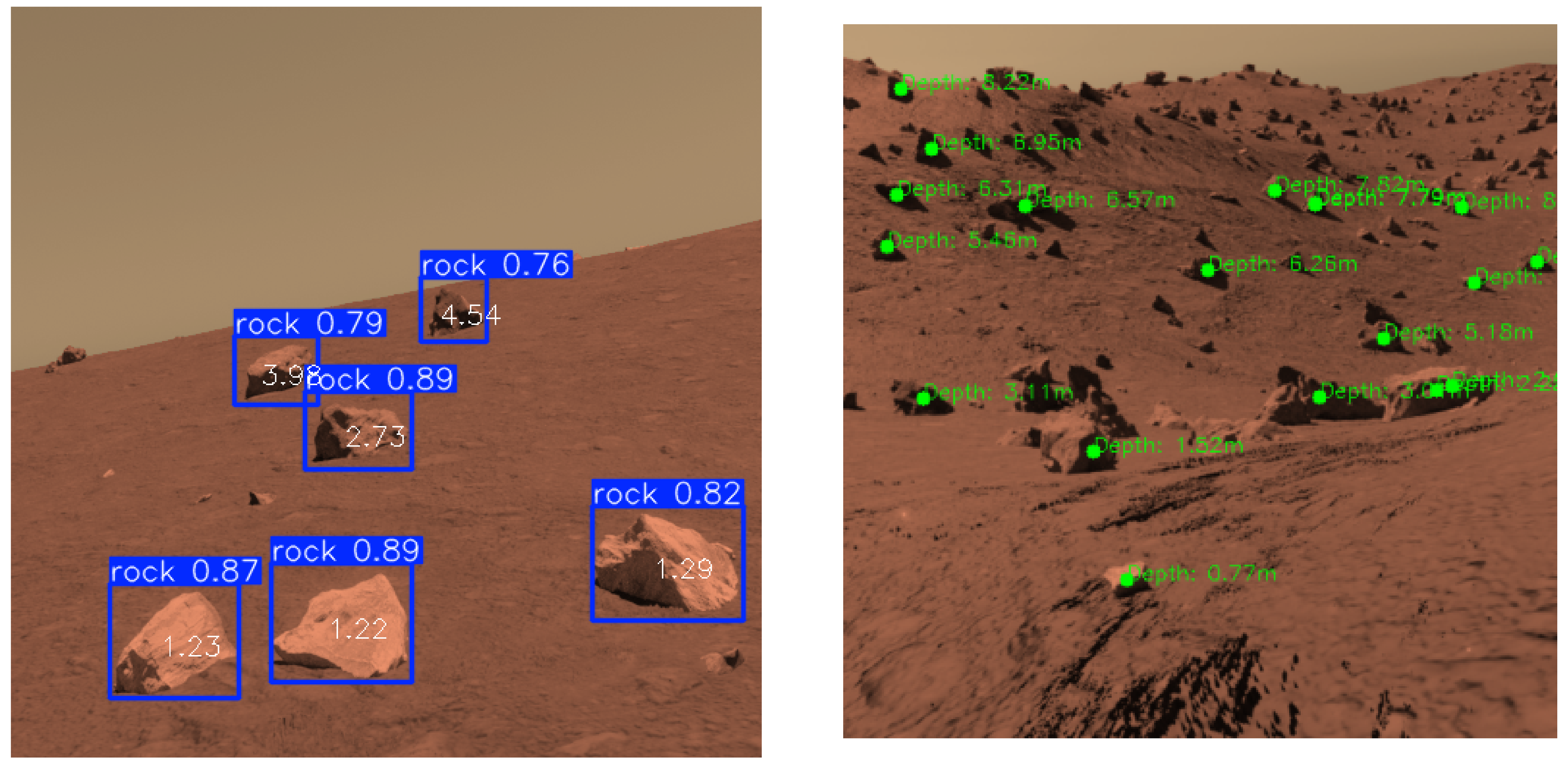

Figure 4. Output images predicting the relative depth of rocks

ROS and Simulation of URDF file using Gazebo

ROS(Robot Operating System) is a framework which is used here to integrate the components(eg.- ZED Camera, microprocessor, motor encoders,etc.) of the rover with each other. Simulation of the rover is shown below which is shown by building up a URDF file.URDF is based on XML, so everything is represented as a series of tags, which can be nested. The main tags we have used are the robot tag and XML declaration, link tags, and joint tags.Whenever we want to use the URDF we need to run the xacro software over the files first, which will process them into a single, complete URDF file.In the visual structure, we added colors, a base link, a chassis connected to the base link through a fixed joint, drive wheels connected through a continuous joint, and caster wheels.

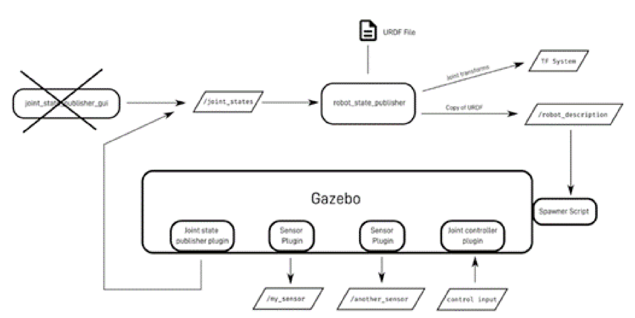

Figure 5. Interactions between Gazebo, ROS components, and URDF for simulating a robot.

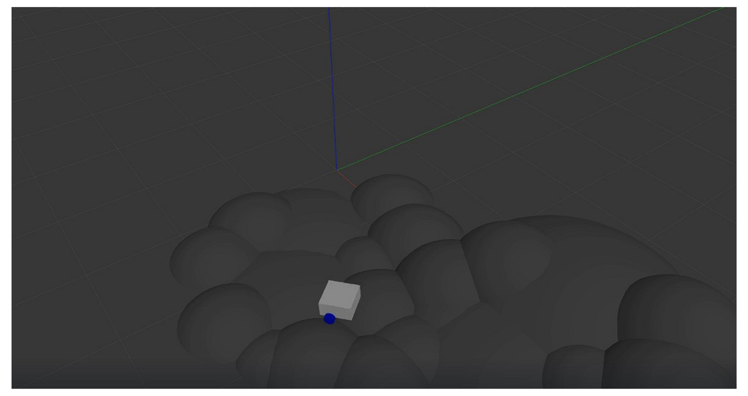

Figure 6. Simulation of the robot in Gazebo

We incorporated collision and inertia for physics calculations and simplified the process using macros.We used Gazebo to simulate the robot that can be controlled, read sensor data, and provide feedback on actuator states.Gazebo doesn't use URDF files; it uses SDF. Additional gazebo-specific information in URDF is added through gazebo tags, which are used to create SDF files and interact with ROS through plugins.

Implementation of SLAM(Simultaneous Localization and Mapping) with ZED Camera

The SLAM (Simultaneous Localization and Mapping) module in ROS is a powerful tool that enables robots to create a map of an unknown environment while simultaneously keeping track of their location within that map. This module combines data from various sensors such as LIDAR, cameras, and IMUs to build and update the map in real-time. ROS provides several SLAM packages, like Gmapping, Hector SLAM, and Cartographer, each with its strengths and suitable applications. These tools are essential for autonomous navigation, allowing robots to understand and interact with their surroundings effectively.

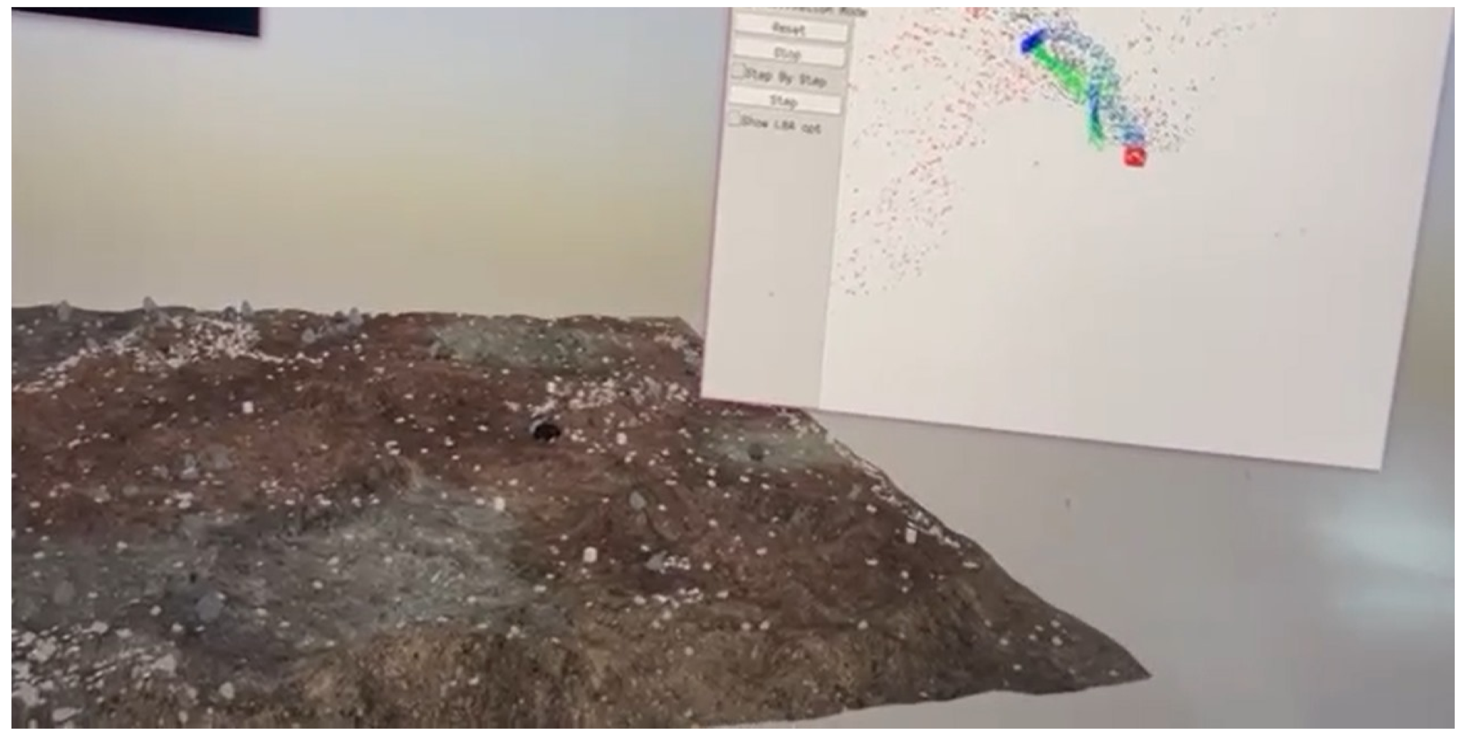

Figure 7. Visual Implementation Using SLAM