Desertification Land Sea-buckthorn Planting Robot Based on YOLOv5 Algorithm

Yi Zhang, Mengmeng Ma, Tao Gong, Yahui Pei*
College of Electrical Engineering, Southwest Minzu University, Chengdu, China
**Corresponding Author.*

Abstract: Desertification poses a serious threat to both humanity and ecosystems, and planting sea-buckthorn can help prevent wind erosion, maintain soil and water, and play a crucial role in improving ecological conditions. Traditional methods of planting sea-buckthorn are characterized by high labor costs and lengthy time cycles, making them inadequate for the demands of large-scale afforestation. To address this issue, this paper presents a tree-planting robot capable of planting sea-buckthorn on desertification land. The robot is equipped with a self-developed integrated mechanical arm for grasping and planting, utilizing SLAM mapping and RTT path planning for autonomous navigation and optimal route planning. Additionally, it features a vision recognition system based on deep learning this tree-planting robot effectively reduces labor, saves time and economic costs, and contributes to the improvement of land desertification.

Keywords: Sea-buckthorn Planting Robot; Integrated Mechanical Arm for Grasping and Planting; SLAM Mapping; Vision Recognition

1. Introduction

Land desertification is becoming increasingly severe, significantly impacting the ecological environment and necessitating large-scale afforestation efforts to improve the ecosystem. This year, the National Forestry and Grassland Administration has formulated and implemented the "Three-Year Action Plan for Large-Scale National Afforestation (2023-2025)", proposing an annual afforestation target of no less than 100 million mu (approximately 16.5 million acres), with at least 20 million mu (approximately 3.3 million acres) dedicated to artificial afforestation. Sea-buckthorn , which thrives in high-altitude areas with gravel or sandy soil, as well as saline-alkali land, exhibits high heat and cold resistance and wind erosion resilience. Therefore, it can significantly increase afforestation area and quality for greening desertification land. Furthermore, sea-buckthorn fruits are rich in vitamin C, making them a highly nutritious wild fruit and a key species for the industrial development of desertification land. With a relatively long lifespan of around 15 years, sea-buckthorn matures from seedling to fruit-bearing in just five years, making it suitable for planting in desertification areas. Traditional manual tree planting in desertification land is inefficient, slow, and labor-intensive. Therefore, there is a need to develop a tree-planting robot [1] [2] capable of planting sea-buckthorn in desertification land. This robot employs an automatic drilling mechanism and an integrated mechanical arm to plant sea-buckthorn seedlings into the drilled holes [3], completing the soil covering process. To ensure the quality of the planting process, the robot utilizes the Yolov5 deep learning-based object detection algorithm [4] to check if the planted sea-buckthorn seedlings are upright and if the soil covering is satisfactory. In case of deviations, the information is transmitted to the control system for adjustments. Additionally, the robot achieves autonomous navigation [5] [6] [7] and path planning [8] through three-dimensional positioning based on a depth camera. The emergence of the sea-buckthorn planting robot for desertification land presents a promising solution to address these challenges. These robots incorporate advanced technologies, enabling autonomous tree planting and aiding in the restoration of desertification land ecosystems at low production costs, thereby enhancing economic benefits.

2. Mechanical Design

2.1 Mechanical Arm Design

To enable the robot to effectively plant trees in desertification land, an integrated mechanical arm for grasping and planting is designed[3], as shown in Figure 1,. The device consists of a six-degree-of-freedom mechanical arm and a grafting device. The six-degree-of-freedom mechanical arm is composed of multiple joints, each capable of free rotation or movement. Typically, a six-degree-of-freedom mechanical arm consists of a base, two rotary joints, and three linear motion joints. The base is fixed to the main body of the robot, and the rotary joints allow the mechanical arm to rotate in both horizontal and vertical directions. The linear motion joints enable the mechanical arm to move in three-dimensional space. The highlight of this mechanical arm is its dual functionality of both gripping/clamping tree seedlings and soil covering, as depicted in detail in Figure 1. This design significantly optimizes the structure of the sea-buckthorn planting robot, reducing costs while effectively achieving sea-buckthorn planting.

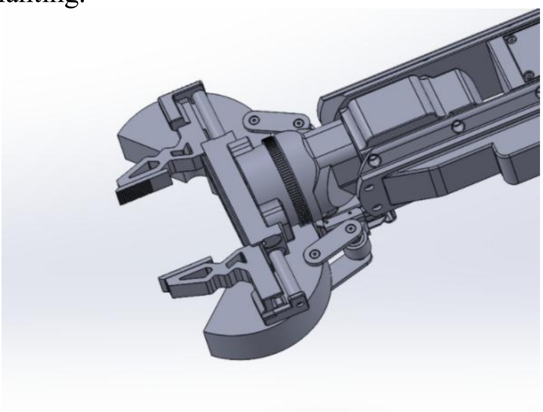

Figure 1. Modeling Diagram of Mechanical Arm

2.2 Chassis Design

Considering the soft and complex nature of desertification land, a metal tracked chassis is chosen for the robot. It is driven by four DC motors, providing ample power for the vehicle's movement. The tracked driving method, compared to wheeled driving, exhibits stronger adaptability to the ground, enhanced climbing ability, smoother movement, and significant advantages in handling complex and heavy-duty operations. Additionally, the large ground contact area of the tracks increases stability. In soft soil conditions, where sinking is shallow, the probability of malfunctions is greatly reduced. The modeling diagram of the chassis is shown in Figure 2.

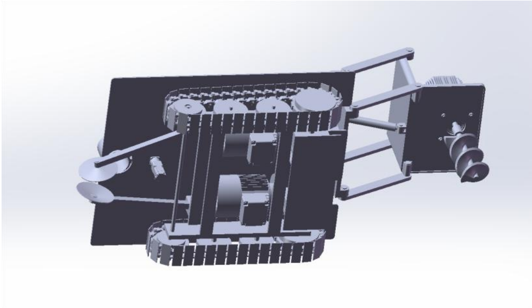

Figure 2. Chassis Modeling Diagram

3. System Design

The robot utilizes ROS (Robot Operating System) as the main control system and STM32 microcontroller as the control center. The ROS main controller communicates commands to the STM32 microcontroller through serial communication, instructing the mechanical arm to reach specified coordinate positions. The robot then utilizes the drilling end to bore holes, lifts the mechanical arm, switches to the gripping end, picks up sea-buckthorn seedlings from the robot's body, places them into the holes, and levels the soil to complete the tree planting process. The Yolov5 object detection algorithm based on deep learning is employed to recognize the status of the seedlings and output two-dimensional pixel coordinate information for their endpoints. As the seedlings have a relatively large volume, the project independently designed a two-degree-of-freedom camera mounting device that can rotate left and right by 180 degrees and up and down by 90 degrees. This allows for multi-angle capturing of images, addressing issues related to obstruction. To achieve autonomous navigation, SLAM (Simultaneous Localization and Mapping) three-dimensional mapping and RRT (Rapidly Exploring Random Trees) path planning are employed.

3.1 YOLOv5 Recognition Algorithm Target Detection

YOLOv5 is a deep learning algorithm designed for object detection. [4] This algorithm represents the latest version in the YOLO series and is widely applied in the field of computer vision. Object detection is a crucial task in computer vision, aiming to accurately identify objects of different categories from images or videos and determine their locations. YOLOv5 is capable of achieving real-time object detection with significant improvements in both accuracy and speed.

The following are the main features and principles of the YOLOv5 algorithm:

(1) Single-stage Detector: YOLOv5 is a single-stage object detector, meaning it can perform both object detection and classification in a single forward pass. It is more efficient compared to traditional two-stage detectors.

(2) Network Architecture: YOLOv5 adopts a backbone network structure called CSPDarknet53. This network structure employs a method called Cross-Stage Partial (CSP) connection, which improves feature extraction without increasing the number of parameters.

(3) Multi-scale Prediction: To adapt to objects of different scales, YOLOv5 uses the Feature Pyramid Network (FPN) structure. The FPN structure can integrate feature maps from different levels, enabling detection and localization of objects at different scales.

(4) Anchor Mechanism: YOLOv5 uses an Anchor mechanism to generate predefined bounding boxes. These bounding boxes are used to detect the position and size of the targets. By predefining a set of Anchors with different sizes and aspect ratios, YOLOv5 can predict bounding boxes adaptively based on the scale of the targets.

(5) Data Augmentation: To enhance the model's generalization ability, YOLOv5 introduces various data augmentation techniques. These techniques include random scaling, rotation, cropping, flipping, etc., which increase the diversity of the dataset and improve the model's robustness.

(6) Training Strategy: YOLOv5 employs a data augmentation method called Mosaic to improve the model's generalization ability. Mosaic combines four different images randomly to create a synthetic image containing multiple objects. This training method allows the model to better learn the contextual information of the targets.

In summary, YOLOv5 is a fast, accurate, and efficient deep learning algorithm for object detection. Through techniques such as single-stage detection, multi-scale prediction, and data augmentation, it achieves excellent performance in real-time and accuracy. The desertification sea-buckthorn planting robot, using the YOLOv5 algorithm, can more accurately identify the status of sea-buckthorn seedlings and transmit this information to the control system for adjustments. After the mechanical arm completes the soil covering, the algorithm can quickly recognize and judge the adequacy of the soil covering. The overall flowchart of the YOLOv5 algorithm is illustrated in Figure 3.

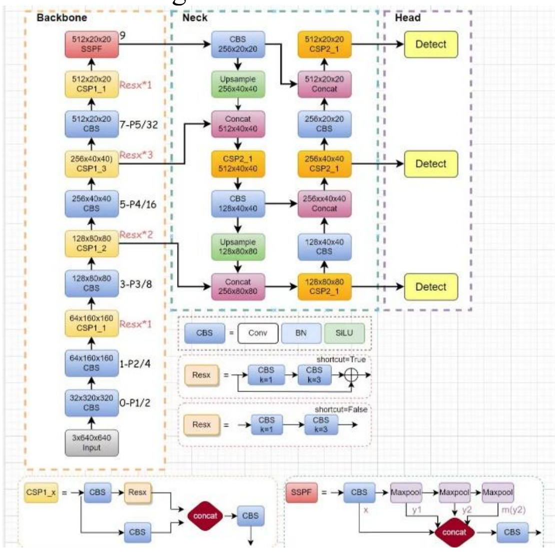

Figure 3. Overall Network Structure Diagram of YOLOv5 Algorithm

3.2 Tree Planting Site Localization

The localization module primarily involves converting two-dimensional pixel coordinates into three-dimensional spatial coordinates. Initially, the depth camera captures color and depth images simultaneously, utilizing stereo vision to calculate depth through infrared sensors and emitters. The OpenNI_ROS2_SDK library is then used to align the depth stream with the color stream, obtaining the depth value corresponding to the target object's two-dimensional pixel coordinates as output by the recognition module. Subsequently, using the depth information and the intrinsic parameters of the camera's color sensor, the two-dimensional pixel coordinates of the color image are converted into three-dimensional spatial coordinates with the color camera as the reference frame. Finally, the TF library is employed to transform the coordinates from the color camera's reference frame to the mechanical arm's reference frame based on their relative positions. The resulting coordinates are then sequentially transmitted to the mechanical arm control system through a scheduling strategy. Based on the recognized center pixel coordinates and depth values of the sea-buckthorn seedlings, the three-dimensional spatial coordinates of the seedling's center are obtained through coordinate transformation. The mechanical arm is then controlled to reach the specified coordinate position, and the end effector is used to perform operations such as grasping and planting on the target sea-buckthorn seedling.

3.3 SLAM Mapping

SLAM (Simultaneous Localization and Mapping) is an algorithm that simultaneously performs localization and map construction. [5][6][7] The principles of SLAM mapping can be broken down into the following steps:

Sensor Data Acquisition: The desertification sea-buckthorn planting robot, equipped with sensors, gathers environmental information in the surrounding area. This includes data such as distance and angle measurements from LiDAR, as well as image information from cameras.

Data Preprocessing: Sensor data undergo preprocessing steps such as denoising and filtering to enhance data quality, enabling subsequent algorithms to handle the data more effectively.

Feature Extraction: Using feature extraction algorithms, differences between sensor data from the current and previous moments are extracted as features. For example, edge information of objects on the map can be extracted from LiDAR data.

Feature Matching and Localization: The features collected at the current moment are matched with previously stored features in memory. Registration algorithms are used to calculate the robot's current position and update the map.

Backend Optimization: Non-linear optimization is applied to refine the results of localization and mapping, aiming to improve accuracy, reduce the impact of noise, and optimize the quality of the map, etc.

Loop Detection: Over a certain period, the robot may pass through the same area multiple times. During this time, the previous map and localization results are examined to determine whether corrections are needed for the map and position information.

In summary, the principles of SLAM mapping rely on sensor data acquisition. Through processes like feature extraction and matching, the robot's position and map are continuously updated and calibrated in real-time. This allows for the simultaneous achievement of localization and map construction functions. The algorithm finds wide applications in fields such as unmanned vehicles and robotics.

3.4 RRT Path Planning

RRT (Rapidly-Exploring Random Tree) path planning algorithm is a tree-expanding algorithm that efficiently navigates through complex environments. [8] It is a randomized strategy for planning, continuously expanding the tree in a random manner. Below, we will provide a detailed explanation of the implementation principles of RRT path planning.

The RRT path planning algorithm involves two key concepts: the tree structure and random samples. The algorithm starts by randomly selecting a starting point, creating the initial node in the tree, which serves as the root node. Subsequently, the algorithm generates tree nodes using random samples until it reaches the goal point or meets the planning time limit. During each tree expansion, the RRT algorithm randomly selects a goal point and grows from an existing node to a new one.

In RRT path planning, it is essential to ensure the diversity of node distribution, sufficient coverage of the area, and the minimization of the path. Therefore, the process of selecting nodes in the tree must meet two requirements: firstly, nodes should be randomly distributed within the designated area; secondly, nodes should have appropriate distances from existing nodes to ensure that the branches of the tree are neither too dense nor too sparse.

During the tree-building process, random samples are taken from the environment. If a sample point is not feasible with some existing nodes in the tree, it is skipped, and another sample point is generated. If the sample point is feasible, the nearest tree node is found, a new path is connected between them, and the new node is added to the tree, continuing until the goal point is reached.

In practical applications, the RRT path planning algorithm encounters certain issues that require refinement and optimization. For instance, there are concerns related to speed and efficiency. The randomness in the node distribution of the RRT algorithm may result in longer search times compared to other algorithms, prompting the need for state compression techniques to address such issues. Additionally, the problem of local optimal solutions in the algorithm requires human intervention, such as adding random sample points to the boundaries of the exploration area or stopping area.

In summary, the RRT path planning algorithm is based on random sampling and tree branch expansion. Although the implementation of the algorithm is relatively simple, it can achieve high levels of efficiency and planning quality. Through improvements and optimizations in practical applications, the RRT algorithm, in conjunction with SLAM mapping, can fully enable the sea-buckthorn planting robot to navigate autonomously in desertification land, accurately completing the sea-buckthorn planting process.

3.5 Control of the Robotic Arm Movement

To achieve precise drilling at specified tree planting locations with the robotic arm, it is essential to obtain the relative positional relationship between the mechanical arm and the target point. Data obtained from depth cameras and LiDAR represent the relative positional relationship between the sensors themselves and the surrounding environment. The TF library is then used to transform this information into the relative positional relationship between the mechanical arm, drilling, and the target point.

We plan to establish coordinate systems for each sensor and actuator of the robot, storing this coordinate information in ROS for managing coordinate system transformations using the TF library. This enables the management of the robot's changing posture and position information in three-dimensional space over time. It can track changes in coordinate systems during robot operation, merging known information to ensure coordination between components during motion. Moveit motion planning algorithm library will be utilized for motion planning. During the object movement, Moveit uses coordinate system information provided by TF and the state information of sea-buckthorn seedlings provided by the camera. It processes data about the robot and the surrounding environment, calculates the kinematic equations for each robot joint, thereby determining the robot's motion trajectory. It then decides the next grasping or planting action for the robot to achieve the desired final goal. The control flow of the motion mechanical arm is illustrated in Figure 4.

4. Experimental Data

By obtaining publicly available seedling status datasets online and capturing various states of sea-buckthorn seedlings through physical photography, all acquired images are filtered. The dataset is then enhanced through operations such as rotation, mirroring, cropping, etc. The final goal is to achieve a dataset size of 10,000 images with a training accuracy of 98.3%. The accuracy curve is depicted in Figure 5.

Figure 4. Control flow chart of mechanical arm

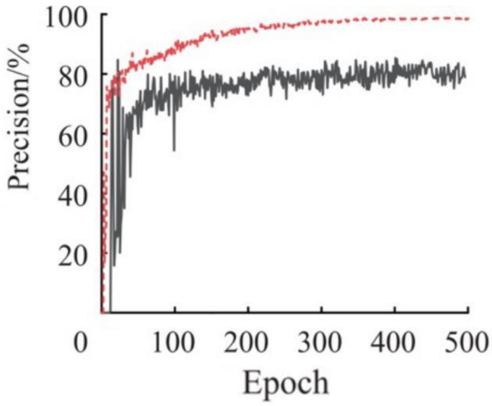

Figure 5. Accuracy Curve

5. Program Debugging and Optimization Iteration

To implement functions such as motion planning and path tracking for the desertification tree-planting robot in the program and ensure that the mechanical arm can move along an approximately ideal trajectory, it is necessary to debug and optimize the system. Based on various scenarios the robot may encounter during tree planting, optimization iterations are performed on the robot's algorithms, mechanical structure, and other aspects. This involves adding some context-aware code, such as obstacle detection, distance calculation, etc., to prevent mutual interference and collisions among the mechanical arm and other obstacles.

6. Summary

This paper addresses the issue of sea-buckthorn planting in desertification land and proposes the design of a tree-planting robot based on ROSE control and deep learning visual recognition, incorporating significant innovations in the mechanical structure. Firstly, a tracked motion chassis is employed to adapt to a wider range of terrain conditions. Secondly, for the grasping and planting of seedlings, a self-developed integrated mechanical arm for gripping and planting is utilized. Finally, during the planting process, autonomous navigation and positioning at planting locations are achieved using SLAM and RRT. The main conclusions drawn from this study are as follows:

(1) Based on the size of sea-buckthorn seedlings and the topographical features of desertification land, a prototype of the desertification tree-planting robot was designed. This includes a tracked motion chassis and a storage device more suitable for sea-buckthorn seedlings, facilitating a smoother planting process;

(2) The YoloV5 object detection algorithm based on convolutional neural networks was utilized to train on a dataset of photos capturing the status of sea-buckthorn seedlings in the natural environment and photos indicating qualified soil coverage during planting. The final training resulted in a dataset of 10,000 images with an average recognition accuracy of 98.3%;

(3) SLAM algorithm and RRT path planning were employed to enable the robot to automatically navigate through obstacles during its movement, enhancing the stability of the robot's operation and making the entire tree-planting process more towards full automation;

(4) The hardware and software structure of the tree-planting robot was designed, and the entire robot was assembled. Field experiments of sea-buckthorn planting were conducted, resulting in a survival rate of over 80% for sea-buckthorn seedlings. Therefore, the robot demonstrates high reliability.

References

[1] Ma Haoyin, Han Xiao, Li Xinyu, Lin Han, Su Miao. (2021). Innovative Design Analysis of Desert Automatic Tree-Planting Robot, China Plant Engineering (21), 86-87.

[2] Li Qiang, Zhao Jinyu, He Bowen, Zhang Zhaoyun. (2023). Design of a Robotic Arm-based Desert Tree-Planting Device. Xinjiang Agricultural Mechanization (05), 12-13+20.doi:10.13620/j.cnki.issn1007-7782.2023.05.003.

[3] Yang Deqing. (2023). Research on Optimization Control of Robotic Arm Based on Multi-Agent Reinforcement Learning Algorithm. In Proceedings of the 3rd International Conference on Innovative Talent Cultivation and Sustainable Development in 2023 (pp. 444-447). School of Information and Electromechanical Engineering, Zhengzhou Business University;

[4] Zhou Xueyou, Wang Xiyuan, Liu Yongchang, Liu Yawen, Qin Lei. (2023). Vertical Displacement Measurement Method Based on YOLOv5 Object Detection Model and Vision. Geospatial Information (12), 25-28.

[5] Liu Liyuan. (2022). Research on SLAM Algorithm for Intelligent Vehicles in Dynamic Scenes Based on Multi-Sensor Fusion (Master's thesis, Jilin University). https://link.cnki.net/doi/10.27162/d.cnki.gjlin.2022.004436doi:10.27162/d.cnki.gjlin.2022.004436.

[6] Wang Yujie, Guo Hang, Yu Min, Zeng Xiang, Chen Xin, Shi Liang, Zhu Chen. (2021). Research on Laser SLAM Algorithm Based on Three-Dimensional Lidar. (eds.) In Proceedings of the Collection of Satellite Navigation and Positioning Technology (2021) (pp. 28-31). Nanchang University; Jiangxi Normal University;

[7] Liu Nian. (2020). Research and Implementation of Visual-based Multi-Robot SLAM Algorithm (Master's thesis, University of Electronic Science and Technology of China). https://link.cnki.net/doi/10.27005/d.cnki.gdzku.2020.004423doi:10.27005/d.cnki.gdzku.2020.004423.

[8] Gong Hao, Tan Xiangquan, Li Jiaxin, Wu Qingwen. (2024). Research on Path Planning of Mobile Robots Based on Improved RRT Algorithm. Modular Machine Tool & Automatic Manufacturing Technique (01), 19-24. doi:10.13462/j.cnki.mmtamt.2024.01.005.