

Autonomous Navigation and Environment Mapping for Robot Weed Detection

Jisha George
School of Computer Science
University of Lincoln
Lincoln, United Kingdom
13488071@students.lincoln.ac.uk

Abstract— Weeding a farm can be very labor-intensive as it is either done by tractors or hand. Weed control can be very expensive ranging from \$50 to \$1000 per acre (the most expensive being for organic farms), by using robots for weed control the cost of maintaining the farm is significantly reduced, as the need for manual labor is decreased. The goal of this paper is to develop an autonomous robotic system that navigates through a dynamic environment, while detecting and spraying the weeds with herbicide. The navigation system avoids collision with the objects in the environment and the image processing system picks up weeds from the rows.

Keywords— *Autonomous Robot, Navigation, ROS, Sensors, Weed Detection*

I. INTRODUCTION

Weeding is a laborious job particularly for humans. Identifying weeds from the plants can be a difficult task, especially when the weeds and the plants have similarly shaped leaves, and/or colour. Once the weeds have been identified the person would have to manually spray the weed with herbicide or carefully remove the weed from the ground, both of which can be very time consuming and not at all profitable for the farmer [1]. By introducing robots to do both the weed identification process as well as spraying herbicides, a lot of time can be saved and the overall cost would be cheaper for farmers.

The aim of the project was to create a system that could autonomously navigate a robot through a dynamic farming environment, while detecting weeds that were growing between the rows of crops and spraying the weeds with herbicide.

The objectives of the project were as follows:

1. To effectively detect weeds in the ground using image processing.
2. To make the robot autonomously navigate through the environment.
3. To implement an object avoidance system to ensure the robot does not collide into other objects in the field.

II. RELATED WORKS

A. Image Processing

Weeds are plants that are considered to be undesirable [2]. Depending on the type of weed and the crops that are growing, the image processing system needed, to distinguish weed from crop can vary. If the weed has a different colour to the crops being grown then a simple filtering process would be enough to distinguish the crop from the weed, as the saturation, hue or intensity of the crop

would differ to the weed. However, issues occur when the weeds and the crops are indistinguishable by colour.

Reference [2] used a thresholding value for minimum edge frequency to isolate weeds that have different shapes to the crop, but this method of image processing does not detect small weeds that may be separate in the field as it will not mean a threshold value. This method can be used for weeds that have similar edge frequencies as the crop, but the threshold value had to be changed very carefully to only detect weeds and not the crop itself.

Using image processing on real-time camera feed can be very computationally heavy, as the image processing is done on the go which can slow-down systems meaning there can be a delay between finding the weed and spraying or some weeds can be missed due to the image processing not completing before the robot went over the weed.

B. Navigation

The robot cannot explore unknown environments unless it is given sensors to get information from the environment [3]. Autonomous navigation is challenging, specifically in a dynamic environment which could change the map the robot uses at any point, this can affect localisation of the robot as the new map does not match with the current map the robot is using.

Reference [3] used a navigation stack that was comprised of different ROS packages such as move_base and gmapping for autonomous navigation. During the mapping and navigation, the robot had difficulties with surface difference and identifying objects that had wider bases due to the positioning of the laser.

In [4] the author had used a vision-based obstacle detection and navigation, they had found that using stereo vision was appropriate for detecting obstacles in a field, however one of the problems they had was that obstacles need to differ in appearance. The results from this paper cannot be considered as conclusive as they had only tested on static obstacles.

C. Spraying

Spraying herbicide on weeds are the quickest and cheapest way of dealing with weeds, however, herbicides have a negative impact on the environment. To minimise the impact it is best to directly spray the herbicide onto the weed and not just cover a general area.

Reference [5] had used a smart herbicide sprayer that could locate the weeds in a field and would spray the herbicide directly onto the desired spot.

III. METHODOLOGY

A. Image Processing

To achieve the first objective of having a weed detecting system, a simple image processing pipeline was implemented. When implementing the system, a weed was considered to be any plant in the row that was not the desired crop. The image processing portion of the system was implemented first, as this was the core aspect of the robot, since the robot would have to detect weeds before it can spray the weeds with herbicide.

In the gazebo environment there were three types of crops being grown (basil, cabbage and onion), each with different types of weeds growing between the plants. In the basil patch, the colour of the basil was distinctively brighter than the weeds that were growing in the rows. In the cabbage row, the colour of the weed and the colour of the cabbage didn't have enough variation to distinguish between the two, however the sizes of the two plants differed, as cabbages in the row were a lot bigger than the weeds that were growing around it. Finally, in the onion row, there were two distinctive regions, one region had little to no weeds in the row. However, the second region was covered in weeds, making them almost indistinguishable from the onion crops. The methodology will only tackle how the weed detection was done for the cabbage and basil.

To combat weeds which have a distinct colour to the desired plants, an inter-plant weed detection method was implemented. This method of detection can identify weeds that grow between the plants [2]. The method was implemented by extracting the image from the camera attached to the robot, filtering out the hue of the basil and the ground from the image, leaving only the weeds. Once an image with only weeds was obtained, the image was filled to remove any holes that may have appeared during the filtering, afterwards the image was eroded with a kernel size of 1 by 1 to remove small pixels that could have been identified as a weed by the system. Fig 1 shows the camera view of the robot, Fig 2 shows the final mask output.



Fig. 1. Basil Image with weed growing behind leaves



Fig. 2. Mask detects weed behind leaves

For times when a simple colour filtering process is not enough to isolate the weeds from the crops, other morphological methods of image processing must be used. Once again the image is extracted from the camera, but this time the image is converted into a binary image, and filled to remove holes. Afterwards, the binary image was eroded with a kernel size of 25 by 25, this was done to remove all the weeds in the image leaving only the cabbages in the image. Once the eroding process was finished, the image was reconstructed using the original image and the new mask, the reconstruction process builds a new image by putting the original image under the eroded mask. The final output was achieved by removing the reconstructed image from the cabbage only mask leaving a mask with only weeds. This was an inter-row method of weed detection [2] as weeds that were growing between the plants were classed as crops or ignored. Fig 3 shows the camera view of the cabbage row and Fig 4 shows the final output of the mask.



Fig. 3. Cabbage row, circles show weeds growing between crops

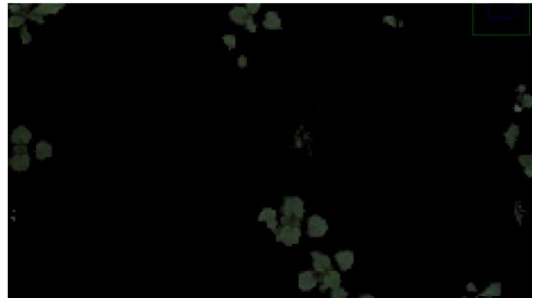


Fig. 4. Mask fails to detect weeds around cabbage

Once the masks for both the basil and the cabbage crops were created, the sum of both the masks were checked to see which one was greater, this was done because the cabbage mask recognized the basil crops as weeds due to the size of basil being smaller than the size of cabbage. A similar problem occurred with the basil mask identifying cabbage as a weed, this was because the basil mask was done on colour filtering and the colour of the cabbage and the colour of the weeds did not have a noticeable change. The mask with the smaller sum was passed into a function which would align the robot to spray the weed.

B. Navigation

To achieve the second and third objectives of having the robot autonomously navigate through the environment while avoiding obstacles the package MOVE_BASE was implemented. Move_base has a built-in marking and clearing system that was used as a mapping system for the robot so that it would create a costmap of the environment surrounding the robot. The move_base package sends move commands to the robot to move safely in the environment while avoiding obstacles in the environment both stationary and moving [3].

A launch file was used to run the move_base package, the launch file runs move_base by getting its configurations from four specified YAML files which states the parameters needed for move_base such as the robot's footprint, what sensors should be used for marking, tolerances for movement etc. The YAML files used for the move_base package are the costmap_common_params, the global_costmap_params, the local_costmap_params and the planner. These four files are used for the path planning and autonomous navigation of the robot [3].

Some aspects of the environment were not being detected by the move_base marking system or the GMapping package that was used to create a map of the environment while the robot was moving. The reason why gmapping did not work was because of the height of the laser sensor. The position of the laser sensor can cause many problems when it comes to mapping and navigation, on the Thorvald robot the laser scan was the same height as the base of the robot, this meant that any objects that were lower or higher than the base of the Thorvald robot were undetected by the robot, which could lead to the robot colliding with undetected objects. The issue was countered by using a package called VELODYNE_LASERSCAN, which worked by extracting a ring of velodyne points from the robot and converting pointcloud data into laserscan data. The velodyne laserscan then had to be filtered so that it would only detect objects that were smaller than average distance from the chosen velodyne ring, otherwise the velodyne will detect the floor as an obstacle.

As seen in Fig. 5 in the gazebo world there is a pallet that is low on the ground, Fig. 6 shows that the laserscan (red) could not detect the low object, however the velodyne (yellow) did detect the pallet.



Fig. 5. Gazebo Simulation World

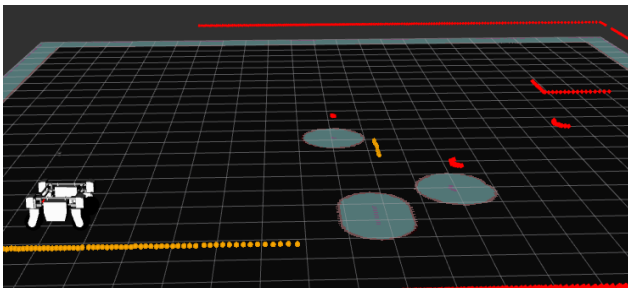


Fig. 6. RViz showing three objects detected by sensors

C. Spraying

The second part of the first objective was to have the robot spray herbicide onto the weeds. Using herbicide can

have negative impacts on the plants and the soil. The easy way to lessen the consequences of herbicide is to spray directly onto the weed, this reduces the amount of herbicide wasted [5].

The spraying function would only activate when the sum of the mask was greater than a specified threshold value. If that was the case then the system would take a snapshot of the mask, if there is nothing in front of the robot that is less than 1 meter then the robot will move forward one meter, which is the distance of the sprayer from the camera, once the robot assumes the sprayer is aligned with the weed it will spray the weed and move backwards to the previous position and align itself to the row. Once it has finished aligning itself, the robot will proceed to move along the path, until it finds another weed, repeating the process until the robot has reached the end of the path.

As a result of the implementation the spraying system of the robot had a reactive behaviour that would override any other instructions that were given to the robot until the robot has finished weeding the row of crops.

IV. EVALUATION

A. Image Processing

Due to the morphological image processing framework that was used to create the filtered masks for the weeds, the system will not identify crops as weeds in the respective masks. However, as mentioned before, because of the basil filtering using an inter-plant system there is a possibility of the weeding system spraying the crop with herbicide along with the weeds as a result of detecting weeds that grow between the row of crops, meaning that the robot will go towards the plant to spray it with herbicide.

The opposite can be said to the weeding process in the cabbage row. The cabbage row has a higher chance of missing a weeding because the cabbage filtering uses an inter-row method of image processing. This means that the system will ignore any weeds that can growing between the plant crops.

When the two types of weeding systems were compared, the inter-row method would be more suitable than an inter-plant method, for an herbicide spraying robot. If the system sprays the crop during the process of weeding, it can contaminate the crops with chemicals which can cause the plant damage, especially if the herbicide being used is stronger than usual, this could lead the crop to no longer be viable for harvest

B. Navigation

Move_base was used for both the movement of the robot and the mapping of the environment by using the marking option in the YAML file. The marking option created a costmap of all the objects in the environment, which gave the robot information on how close the robot could get to an object before the robot had a collision.

The laser and velodyne were used to both detect objects in the environment and to ensure the robot would not move towards an object when it was within a meter, this function acted as a safety precaution.

Due to the laserscan not being able to detect low or small objects like a pallet or high objects such as a branch from a tree, another option had to be explored. At first a package called POINTCLOUD TO LASERSCAN was implemented. However, it did not work, this may have been due to the velodyne pointcloud received data from multiple rings. There was little documentation on the package which caused an inability to use the package properly and convert the data as required.

A different package called VELODYNE-LASERSCAN was implemented, this package would extract data from a specific ring of the velodyne and convert the pointcloud data into laserscan data. This package worked a lot better than the pointcloud-to-laserscan package and the package allowed options of which ring to extract data from.

A lot of systems were tested for the mapping function of the system. One of the main mapping systems that was considered to be implemented was a package called RTABmap. RTAB (Real-Time Appearance-Based Mapping) was initially going to be used to build a 3D map of the environment, however RTAB uses both camera and lidar to build the map, this was a problem for the Thorvald robot as the robot only has one camera that faces straight down towards the ground.

Another package that was considered for 3D mapping of the environment was a package called LOAM (Laser Odometry and Mapping) velodyne. This is package that creates a 3D map using pointcloud data from the velodyne. However, this system was not implemented due to documentation on the package talked about implementing the system with a bag file. This form of implementation was not suitable for a weeding scenario as the robot would have to record data of all the topics while moving through the environment.

Using the velodyne-laserscan package, the parameters of GMAPPING were changed to use the velodyne scan topic, however this caused an issue with the mapping, where the chosen ring was either too big or too small of mapping. Because of the way gmapping creates a map, when the velodyne scan hit the floor, gmapping would mark that area as a border of the environment instead of unoccupied space, this caused a lot of localisation issues.

With navigation, the move_base package was the best to use especially for movement, as it has the ability to use the map provided as well as build its own costmap to check for changes in the environment. This is suitable for a farming environment because the surroundings in a farm will not always be the same, there will be workers walking around that the robot has to be careful to avoid or objects that are placed on the floor.

The marking system from move_base allowed for the detection of both moving and stationary objects due to the YAML file allowing for multiple sensors to be used to create the costmap. Move_base plots the path of a moving object and marks the area as occupied, this helps the robot avoid collision,. However, it also meant that parts of the field were unavailable to the robot.

C. Spraying

The spraying system was done in a very archaic method as there were other options that could have been implemented. A tf system could have been introduced to the weeding process that would have supplied specific coordinates of the detected weeds, this would have been sent to the robot through move-base and sprayed when the position of the sprayer was aligned with the coordinates that were calculated.

REFERENCES

- [1] T. E. Madsen, and H. L. Jakobsen, "Mobile robot for weeding," MSc. Thesis, Dept. Control and Eng. design, Tech. Uni. Denmark, Denmark, 2001.
- [2] M. Latha, A. Poojith, B. A. Reddy and G.V. Kumar, "Image processing in agriculture," Int. Jnl. of innovative Res. in Elect., electronics, instrumentation and control Eng., vol. 2, no. 6, 2014
- [3] S. Zaman, W. Slany, and G. Steinbauer, "ROS-based mapping, localization and autonomous navigation using a Pioneer 3-DX robot and their relevant issues," Saudi Int. Electronics, Communications and Photonics Conf. (SIECPC), p. 1-5, April. 2011.
- [4] D. Ball et al, "Vision - based obstacle detection and navigation for an agricultural robot," Jnl. of field robotics, vol. 33, no. 8, p.1107-1130, 2016.
- [5] A. H. Kargar B. and A. M. Shirzadifar, "Automatic weed detection system and smart herbicide sprayer robot for corn fields," First RSI/ISM Int. Conf. on Robotics and Mechatronics (ICRoM), p. 468-473, 2013,