Intelligent Agriculture Robot for Tea Plantation Preservation - TeaBot

Imalka Gunawardana*, Varnavi Perera†, Thulya Premathilake‡, Gayan Bamunusinghe§, Shashika Lokuliyana¶, Narmada Gamage‖, Rajitha de Silva**
*,§ Dept. of Computer Science & Software Engineering, Sri Lanka Institute of Information Technology
†,‡ Dept. of Information Technology, Sri Lanka Institute of Information Technology
¶,‖ Dept. of Computer Systems Engineering, Sri Lanka Institute of Information Technology
** Lincoln Agri-Robotics, University of Lincoln, United Kingdom
*it19973470@my.sliit.lk,†it20382476@my.sliit.lk, ‡it20265410@my.sliit.lk,§it20011970@my.sliit.lk,¶shashika.l@sliit.lk
‖narmada.g@sliit.lk,**rajitha@ieee.org

***Abstract*—Tea cultivation stands out as one of the most significant export products that contribute to the Gross Domestic Product (GDP). In the previous decades, the substantial workforce gradually transitioned to different occupations. Compared to other crops, tea cultivation demands costly and meticulous upkeep. With a shortage of labor, maintaining tea estates grew tougher, leading to decreased yields. In reaction, estate owners transitioned to cultivating low-maintenance crops. The TeaBot is an advanced robot designed to replace human labor for watering and fertilizing vast tea estates. The TeaBot distinguishes itself by operating in rough outdoor terrain and infrastructure-free navigation in real tea plantations. Given that tea plants demand continuous hydration and nutrients for optimal crop yield, TeaBot plays a pivotal role in enhancing efficiency while reducing water and fertilizer wastage. Mainly four motor-powered wheels move accurately by translating linear and angular velocities using a precise motor control algorithm. An autonomous navigation algorithm was developed using two distinct approaches, which include deep learning-based computer vision and classical computer vision. The selection of the classical computer vision method was predicated upon its notable attributes, including high precision, minimal resource utilization, and optimal efficiency. A deep learning-based stem identification model was trained based on MobileNetV2 architecture to detect where the plant stem meets the ground for efficient hydration of individual plants. This lightweight model achieved 90% detection accuracy. The precise results of stem detection have made the liquid fertilization process more efficient.**

***Keywords*—Precision Agriculture, Computer Vision, Robotics, Machine Learning, Autonomous Navigation**

I. Introduction

The global tea industry's significance, especially Ceylon Tea as a renowned beverage and major export, faces challenges due to a lack of labor in large-scale tea cultivation [1]. The TeaBot robot emerges as a solution, specializing in automated watering and fertilization. A shortage of workers shifting to other jobs has made maintaining tea estates difficult, resulting in reduced yields and impacting exports. In response, TeaBot addresses this labor scarcity by employing advanced automation technologies. It navigates and identifies tea stems through deep learning algorithms and precisely applies liquid fertilizers. The motivation behind TeaBot's development lies in changing workforce preferences, which have created a shortage of labor for traditional tea estate management [2]. This has not only affected exports but also led to cultivating low-maintenance crops, affecting the market. TeaBot's specialties include automation navigation, traversability in rough terrain, and accurate stem identification for precise fertilization. Deep learning algorithms enable accurate navigation and stem recognition. Motor control algorithms convert velocities into wheel actions and stem coordinates into servo motor actions, ensuring precision. When considering TeaBot's implementation, highlighting navigation, deep learning-based stem identification, and path navigation algorithms. The motivation of this work is to emphasize TeaBot's role in sustainable large-scale tea cultivation, addressing labor shortages and optimizing yields. By showcasing path navigation algorithm optimization, the paper underscores TeaBot's accuracy and efficiency, potentially revolutionizing tea estate management and reestablishing Ceylon Tea's global prominence. The key outcomes of this research can be displayed in the following manner:

- Semantic segmentation-based path detection algorithm.
- An algorithm that detects the linkage between the stem and the soil for optimized spraying.
- An autonomous algorithm that synchronizes spraying based on stem detection.

II. Literature Review

TurtleBot3, utilizing a LiDAR sensor to create a map, was employed in navigation tasks for obstacle avoidance using the Robot Operating System (ROS) by [3]. Direct Current (DC) motors are powered and controlled by the Raspberry Pi, which integrates a soil moisture sensor for automatic irrigation, a Passive Infrared (PIR) sensor, and a rainfall sensor [4]. The authors of [5] have developed an Autonomous Mobile Robot that travels on a rail, and the system operates on ROS. Moreover, a four-wheel robot with front-wheel steering employs a Global Positioning System (GPS) and weighs 900 kg [6]. A trajectory tracking system is implemented with a four-wheel skid-steering robot using PID, Raspberry Pi, and

978-8-3503-1770-1/23/$31.00 ©2023 IEEE

Wireless Fidelity (Wi-Fi) [7]. A comprehensive comparison between existing systems and the TeaBot robot structure is provided in Table I.

Several technologies have been suggested to address the autonomous navigation problem of agricultural robots, such as GPS [8], LiDAR [9], and laser-based technologies [10]. Computer vision-based segmentation models have proven to be a successful approach to tackling this problem [11], [12]. Given this consistent pattern, a computer vision-based navigation method becomes suitable for detecting navigation paths within the tea plantation fields. Computer vision-based path detection systems are capable of detecting the path in various environmental conditions [12]. Computer vision-based models perform robustly despite varying field conditions such as shadows, sunlight, weed presence, and discontinuities [11].

Previous studies have demonstrated the potential of techniques, extensively exploring computer vision and deep learning in the detection and tracking of tree and plant stems. However, there is a lack of research specifically focused on identifying the end of tea plant stems (where the stem meets the ground) for optimizing fertilization and hydration. As per the source [13], the localization of stems in sweet-paper plants, while reference [14] into the detection of stems specifically in sugarcane crops. Also, the detection of stems in grape plants [15]. Following the reference [16], it describes X-ray technology for stem detection and lastly, introduces the concept of real-time stem detection [17]. This research requires developing a novel Machine-Learning (ML) model trained on tea stem images and designing a robot control system that uses the end of tea stem coordinates for liquid fertilization. Challenges include variability in tea plant growth patterns, environmental factors, real-time processing, and accurate robot control for the liquid fertilization process.

According to [18], the primary issue with numerous spraying systems is their excessive water usage and inefficiency. These systems lack a precise method for accurately locating plant roots and dispensing appropriate water amounts. Furthermore, a specific mechanism for validating the accuracy of water spraying paths is still absent. Importantly, misting fertilizers are unsuitable for application on tea plants. Another concern is the absence of stabilizing mechanisms for the water spraying nozzles in most of these systems.

III. Methodology

TeaBot is capable of navigating flat and sloped terrain without the need for specially designed tracks. With dimensions of 12 inches in height, 20 inches in length, and 24 inches in width, along with 4 inches of ground clearance, the robot can effectively navigate through the latter parts of tea plant rows. A Raspberry Pi serves as the main controller for the robot's operations, with real-time coordinates provided by computer vision algorithms. The robot's primary objective is to reduce cost, enhance efficiency, and yield a higher return on investment within the tea plantation industry. The proposed system architecture is divided into four main parts as illustrated in Figure 1.

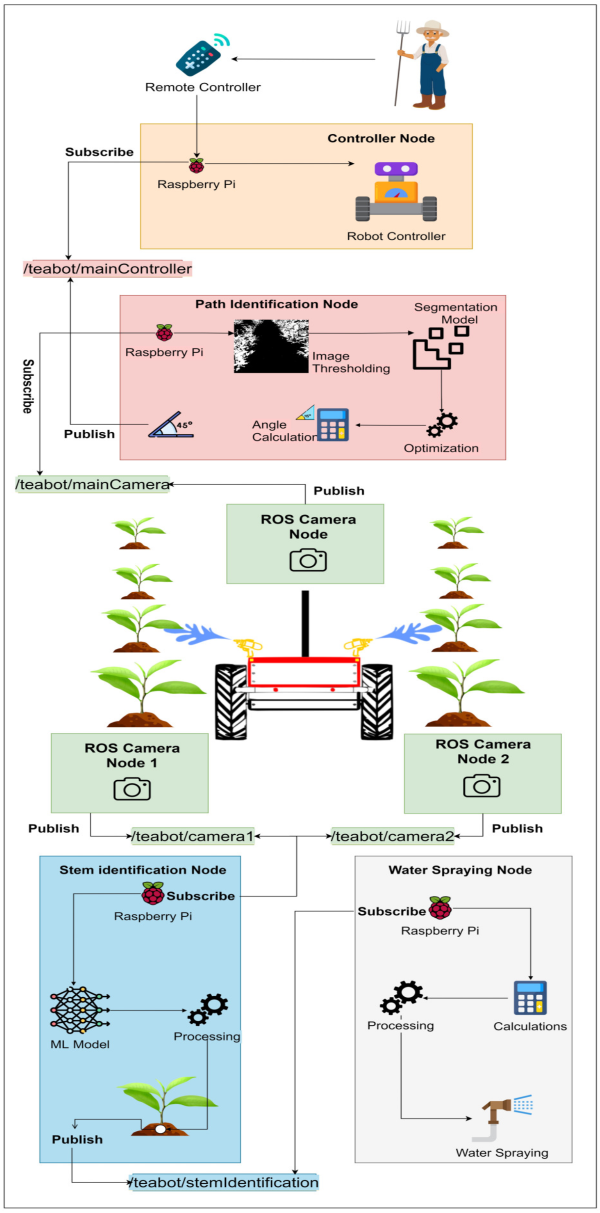


Fig. 1: System overview.

A. Robot Controller

The robot consists of a four-wheel drive electric motor-powered driving mechanism to propel the robot. The main computer is a Raspberry Pi and an Arduino Nano is employed for computational processes. Four motors power all wheels. The Raspberry Pi performs computational processes on top of the ROS Noetic with an Ubuntu Server. It then commands the Arduino Nano through the I2C interface. The robot has two 12V batteries with 8A and another battery with 65A capacity. The 8A battery is used for computational processes, while the 65A battery is utilized for the motors to enhance battery management. The Raspberry Pi features two ROS nodes for the robot controller. One node is responsible for the subscriber, while another node handles the publisher tasks. The publisher manages the remote controller and computer vision coordinates. This publisher node publishes a Twist-type message to interpret linear and angular velocities. Subsequently, the motor control algorithm manages the motors based on the provided linear and angular velocities. The robot controller has hazard identification features. If the robot encounters a hazard that prevents it from moving forward, it utilizes the YOLOv5 algorithm and hardware components to immediately halt its functions, safeguarding the robot components. As reported in [19], YOLOv5 has been enhanced for improved small object detection, specifically for applications in autonomous driving and robotics. The robot is capable of moving forward, and backward, making left and right turns while in motion, and executing skid steering maneuvers to the left and right. The motors possess maximum power to navigate uphill and downhill on steep hills and rough roads. The robot controller is optimized to navigate the wheels according to the linear and angular velocities. The computer vision will provide the angular velocities to the robot controller and the robot can maintain the linear speed for off-road terrain.

B. Robot Arm Design

To effectively water various tea plantation fields, TeaBot's watering system integrates advanced hardware and software components. The system comprises four adjustable watering arms, all sharing the same hardware foundation. Each arm incorporates an Arduino-based 2 degrees of freedom (2-dof) robot arm for water spraying. The robot arm can execute pan and tilt type motion while adjusting itself for the pose of optimal spray toward the detected stem. The hardware configuration also includes an Arduino Nano, which serves as the main controller. Power cables of the Arduino controller are linked to a battery to manage these motors, while water pressure is maintained via a pressure monitoring system [20].

The system incorporates algorithms such as PID and stabilization algorithms to achieve precise control over the arm and accurate water spraying. A dedicated web-based User Interface (UI) allows for manual and auto-mode control. This UI employs a user-friendly palette of two-tone colors, providing essential buttons and functionalities for user convenience. The stability of the robot arm is ensured by a six-axis Internal Measurement Unit (IMU). The robot employs methods to maintain the arm's desired position during motion. This is achieved using the Equation (1), Where θ represents the Stabilized angle, β corresponds to the robot's Tilted angle, and γ indicates the Water spraying angle. The six-axis IMU plays a crucial role in stabilizing the arm.

$$\theta = 90^\circ - \beta + \gamma \tag{1}$$

ROS Framework enables Arduino Nano to communicate with stem detection. The Arduino robot arm is connected to this servo motor. Stabilizing algorithms and the six-axis IMU are attached to the arm to stabilize its y-axis. Several relays are utilized with the water pump to maintain constant water pressure in the arm, and rubber vibration controls are employed to reduce the vibration of the water pump. Figure 2 shows the implementation of the robot arm.

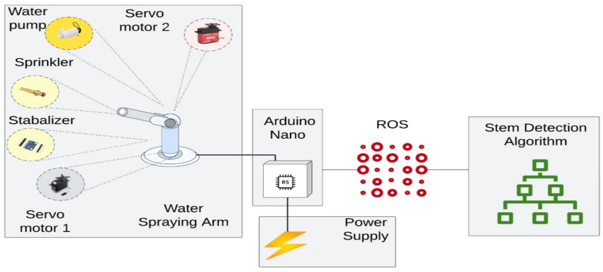


Fig. 2: Root arm controller.

C. Autonomous Navigation

The robot's navigation methodology is addressed through the utilization of a semantic segmentation algorithm. To achieve autonomous navigation, a deep learning-based segmentation model was developed using the U-Net architecture. U-Net utilizes depth-wise separable convolutions and skips connections to adeptly capture features in the provided inputs. The decoder network of U-Net deploys up-sampling feature maps and concatenation with skip connections to reconstruct high-resolution segmentation maps from the condensed feature representations produced by the encoder, guaranteeing accurate navigation path prediction. The construction of this model involved the utilization of web camera videos capturing the navigation trajectory, thereby forming the dataset consisting of a thousand images. During the labeling phase, color differentiation was applied to distinguish the tea plantations and the robot's navigation path. The tea plantations were assigned a white color while the navigation path was represented in black. The differentiation was accomplished using the "excess green" technique, where all pixel values in each frame were modified using the Excess Green Index (ExG) [12] as defined by the Equation (2).

$$\text{Value} = 2 \times (\text{green}) - (\text{red}) - (\text{blue}) \tag{2}$$

Employing the dataset, the U-Net semantic segmentation model was constructed. The resultant mask generated by this model effectively captures the central region of the navigation path. Refer to Figure 4a for an illustrative representation of this process. Upon deployment onto a Raspberry Pi platform, it was discerned that the U-Net model exhibited substantial resource consumption, resulting in relatively lengthy output processing times. An alternative classic computer vision-based navigation algorithm was devised to surmount these challenges. Within this algorithm, the OpenCV Python library facilitated the separation of the tea plantations and navigation path within video frames, again utilizing the Excess Green Index (ExG) for differentiation. The resultant image from this separation was utilized for prediction purposes. Both methodologies adopted an identical approach to determine the positional error (E) of TeaBot. The procedure involved calculating the summation of pixel values vertically for each column of pixels in the predicted binary mask as given in Equation (3). W is the width of the image while Y_x represent the pixel column at each x position. The range of columns with lower summation of pixel values was designated as the path, and within this range, the column with the least pixel value summation denoted the center of the path.

$$E = \left(\frac{W}{2}\right) - argmin\left(\sum_{x=0}^{W} I(x, Y_x)\right) \quad (3)$$

D. Stem Identification

Under this component, the end of the tea stem is identified by analyzing the provided image. Creating a class label for every conceivable combination of (x,y) coordinates of a stem in an input image is not feasible. Unlike classification, which produces a label as output, regression enables us to make predictions of continuous values, covering any real number within a specified range [21]. The objective is to utilize the ResNet50 model, which is based on a Convolutional Neural Network (CNN) architecture, to analyze and identify the final point of the tea stem. The main purpose of this component is to enable the robot to accurately target and apply water and fertilizer to the tea plantations. Without the ability to identify the end of the stem, the robot cannot precisely locate the correct position for watering and fertilization. If the end of the stem is not identified precisely, the robot may apply water and fertilizer to the wrong location, resulting in the tea plant's damage and may lead to inefficient use of resources. As mentioned in reference [22], the ResNet50 architecture is preferred over VGG16 due to its heightened sensitivity to the edges of target objects. Additionally, a comparison between the ResNet50 and VGG16 models reveals variations in accuracy and loss values [23]. However, during deployment on a Raspberry Pi, it was observed that the resource-intensive nature of the ResNet50 model resulted in slow performance. Consequently, a modified variant of the ResNet50 model was fine-tuned, resulting in a more lightweight iteration referred to as MobileNetV2. The network architecture was also adjusted as indicated in Figure 3. The dataset comprises 1000 images. These images have been divided into training and validation sets, with 80% allocated for training and 20% for validation. Furthermore, a separate testing set has been created using images from both the training and validation sets. Prior to model construction, each image was resized to 224 × 224, and all images are in the RGB color profile by default. The initial dataset was pre-processed, and data augmentation was performed due to the limited amount of available data. Finally, the model was compiled and trained using the training and validation data. Once the camera captures a new image using the pre-trained model, the end of the stem is predicted, as illustrated in Figure 7a.

IV. Result & Discussion

Both ROS nodes and the web UI can be employed to control a robot arm. The arm is more precise at the designated angle as it remains stabilized along the y-axis. Water can be precisely sprayed over a distance. The robot arm's x-axis can traverse from 0° to 180°, whereas its y-axis is limited to a range of 0° to 90°. The MPU 6050 gyroscope module remains unaffected by the pump's vibrations and continues to function as intended. This robot arm operates according to the computer vision coordinates provided by the stem detection algorithm, accurately spraying water at the end of the stem. Both methodologies of autonomous navigation, the deep learning-centered computer vision-based algorithm, and the classic computer vision-based algorithm were tested for 100 images. The U-Net has 90% accuracy and it is trained for 20 epochs with a 0.001 learning rate. The model's inference required 5 seconds, with relatively high resource consumption. The test results are depicted in Figure 4a. Performance evaluation was done on the classic computer vision-based algorithm. This algorithm successfully detected the center of the path for 82 out of the 100 images tested. The outcome of this evaluation is illustrated in Figure 4b. The integration of the classical computer vision algorithm's central path prediction with its associated prediction mask has produced compelling color overlay images, as displayed in Figure 5a and Figure 5b. For Figure 5b, it became evident that the achieved results were not accurate. Particularly, towards the end of the path, paths that have a low density of cultivated tea plants. The prediction power is low in these scenarios. Graphs were generated for each tested image, with the x-axis representing the column index of the image, and the y-axis representing the summation of pixel values for each respective column. Accurately generated outputs displayed a mean shifted double Gaussian curve as depicted in Figure 6a. Conversely, in the case of poorly generated images, the characteristic mean shifted double Gaussian curve was not observable, as illustrated in Figure 6b. A comparative analysis of generated graphs, in Figure 6a, the graph's peaks denote rows where tea plants are cultivated systematically, while the central portion represents the navigation path. In Figure 6b these characteristics are not visible as a consequence of the previously discussed scenarios, consequently leading to the generation of imprecise outcomes.

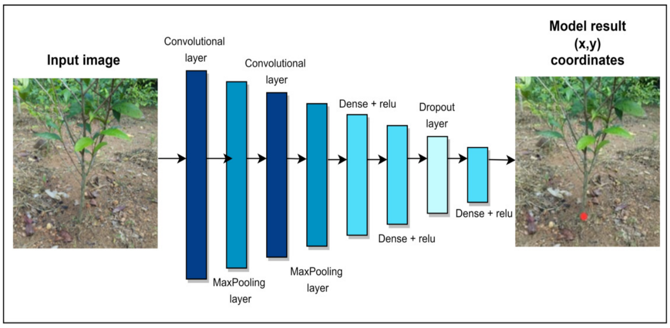
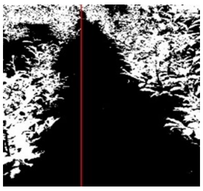


Fig. 3: Stem identification using MobileNetV2.



(a) Deep learning-based approach.

(b) Classic computer vision-based approach.

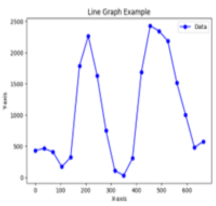
Fig. 4: Test results of path detection algorithms.

(a)

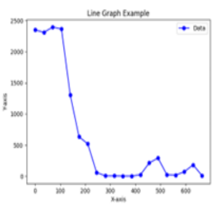
(b)

Fig. 5: Color image with the path detection overlay.

The detection of the end of the tea stem optimizes the precise liquid fertilization process. A thousand frames were created using videos captured through TeaBot, and all frames were meticulously labeled. The end area of the tea stem was denoted by a single dot. The labeled images yielded an Excel file containing the corresponding x and y coordinates. These labeled frames and coordinates were then used to train the image processing model, leveraging the ResNet50 architecture. The model underwent 30 epochs of training with a learning rate set at 0.001, achieving an accuracy of 88%. A streamlined version of the model was developed using MobileNetV2 architecture and trained under the same constraints. Both models were subsequently deployed and tested within the tea plantation field, demonstrating commendable accuracy. The lightweight MobileNetV2 architecture, known for its speed and minimal resource consumption, was utilized to create the solution for detecting the end of the tea stem. The performance metrics for the stem detection algorithm are provided in Table II. The validation dataset demonstrated an accuracy rate of approximately 90% after undergoing multiple iterations of model training. Throughout the training phase, the loss consistently remained quantified at 0.0125, with the validation loss converging to 0.25. The empirical findings of the study



(a)



(b)

Fig. 6: Pixel value summation of the color overlay.

TABLE I: Robots Comparison

Robot	Wheels mechanism	Power supply	Computer	Weight
Thorvald	4WD, 4WS	12V (High power)	Robot control unit	200 Kg
Ragribot	4WD, Skid steering	12V (High power)	Raspberry Pi+Arduino	-
BoniRob	4WD, 4WS	12V (High power)	Intel Atom	-
TeaBot	4WD, Skid steering	12V (Low power)	Raspberry Pi (Python/ROS)	30 Kg

revealed a moderate level of performance when compared to previous works, as accuracy fluctuated between 60% and 90% across iterations in both models. Throughout the testing phase, it became evident that the stem detection algorithm displayed inaccuracies in specific scenarios, as illustrated in Figure 7b. In particular, the algorithm's proficiency in identifying stems within tea plantations of smaller sizes was compromised. The algorithm's effectiveness exhibited fluctuations across a range of environmental conditions, encompassing situations with intense sunlight and shadow effects. In order to improve predictive precision, it was clear that the model's dataset should encompass a comprehensive representation of these diverse scenarios.

The performance of the autonomous spraying algorithm is being evaluated. The algorithm's accuracy increases when the linear velocity of the robot arm is greater than or equal to the velocity of the robot. The formula presented in Equation (4) can be utilized to calculate the linear velocity of the robot arm in centimeters per second, with the angle specified in radians.

$$\text{Linear Velocity} = \frac{\text{Angle} \times \text{Radius}}{\text{Time}} \tag{4}$$

The autonomous spraying algorithm based on stem detection underwent testing within a real-world environment by systematically manipulating the velocities of both the robotic platform and its corresponding arm. The robotic platform was subjected to a series of predefined speeds: 1 cms^{-1}, 2 cms^{-1}, 3 cms^{-1}, 4 cms^{-1}, and 5 cms^{-1}. Concurrently, the robot arm's velocities were set at distinct values of 0.262 rpm, 0.523 rpm, 1.047 rpm, 2.094 rpm, and 3.33 rpm. It has been identified that the algorithm performs accurately except for a few scenarios. Those are when the robot operates at a velocity of 4 cms^{-1} while the arm maintains a speed of 0.262 rpm. Similarly, instances of reduced accuracy occur when the robot operates at a velocity of 5 cms^{-1}, and the arm operates at speeds of 0.262 rpm and 0.523 rpm. It can be stated that as the robot's speed increases, the autonomous spraying algorithm exhibits reduced accuracy.

TABLE II: Stem Identification Model Summary

Model	Accuracy	Precision	Recall	F1
ResNet50	0.88	0.90	0.89	0.91
MobileNetV2	0.90	0.91	0.89	0.92

(a) Precise detection. (b) Flawed detection.

Fig. 7: Results of the MobileNetV2.

V. Future Work

A key focus is addressing challenges with the algorithm for path identification based on segmentation, which has been yielding poor results. A noteworthy observation has been made regarding the error (E) graphs from images that produce unsatisfactory outcomes; notably, these graphs lack a distinct mean-shifted double Gaussian shape which is the usual pattern for images with good predictions. This factor is recognized as a key influencer that leads to outcomes that are less than optimal. To address this issue, the method reshapes the images using mathematical operations to achieve a mean-shifted double Gaussian shape. This should improve the accuracy of path detection outcomes. An opportunity to improve performance lies in exploring larger datasets for pre-training the model. This task may involve using different parts of tea plants grown in varying environments and utilizing stems from tea trees of different ages. To advance the field, it's vital to ensure the spraying arm's stability relative to the robot, which can lead to the motor's shutdown and improved liquid fertilizer spraying.

VI. Conclusion

The TeaBot revolutionizes tea estate management by automating watering and fertilization tasks. With its cutting-edge design tailored for off-road conditions and its ability to navigate autonomously, the TeaBot addresses a crucial need in the tea industry. The evaluation of two navigation approaches yielded intriguing insights. The U-Net algorithm showcased exceptional accuracy at 90%, but its resource-intensive nature prompted the adoption of the classical computer vision method for reliable and efficient autonomous navigation. Developing a lightweight MobileNetV2 model was crucial for identifying tea stem ends. Achieving a commendable accuracy rate of 90%, this model enabled deployment on resource-constrained platforms like Raspberry Pi. TeaBot's precision is evident in its robotic arm, ensuring accurate long-range water spraying with proper alignment. Integrating key components, TeaBot offers a cost-effective solution preserving tea estate integrity and eliminating manual labor. This innovation ensures a transformative impact on the tea industry.

References

[1] "TED Case Study: Ceylon Tea — mandalaprojects.com," http://mandalaprojects.com/giant-project/ceylon-tea.htm, [Accessed 14-08-2023].

[2] C. Rathnayake, B. Malcolm, G. Griffith, R. Farquharson, and A. Sinnett, "Current issues in the farm sector of the sri lankan tea industry," *Australasian Agribusiness Perspectives*, vol. 24, pp. 56–72, 05 2021.

[3] Z. Al-Mashhadani, M. Mainampati, and B. Chandrasekaran, "Autonomous exploring map and navigation for an agricultural robot," in *2020 3rd international conference on control and robots (ICCR)*. IEEE, 2020, pp. 73–78.

[4] M. Arivalagan, M. Lavanya, A. Manonmani, S. Sivasubramanian, and P. H. Princye, "Agricultural robot for automized fertilizing and vigilance for crops," in *2020 IEEE International Conference on Advances and Developments in Electrical and Electronics Engineering (ICADEE)*. IEEE, 2020, pp. 1–3.

[5] E.-T. Baek and D.-Y. Im, "Ros-based unmanned mobile robot platform for agriculture," *Applied Sciences*, vol. 12, no. 9, p. 4335, 2022.

[6] M. Deremetz, R. Lenain, A. Couvent, C. Cariou, and B. Thuilot, "Path tracking of a four-wheel steering mobile robot: A robust off-road parallel steering strategy," in *2017 European conference on mobile robots (ECMR)*. IEEE, 2017, pp. 1–7.

[7] O. Barrero, S. Tilaguy, and Y. M. Nova, "Outdoors trajectory tracking control for a four wheel skid-steering vehicle," in *2018 IEEE 2nd Colombian Conference on Robotics and Automation (CCRA)*. IEEE, 2018, pp. 1–6.

[8] A. SPOORTHI, T. SUNIL, M. KURIAN *et al.*, "Multipurpose agriculture robot using lora," *International Journal of Advanced Scientific Innovation*, vol. 2, no. 1, 2021.

[9] S. Fountas, N. Mylonas, I. Malounas, E. Rodias, C. Hellmann Santos, and E. Pekkeriet, "Agricultural robotics for field operations," *Sensors*, vol. 20, no. 9, p. 2672, 2020.

[10] J. C. Andersen, O. Ravn, and N. A. Andersen, "Autonomous rule-based robot navigation in orchards," *IFAC Proceedings Volumes*, vol. 43, no. 16, pp. 43–48, 2010.

[11] R. de Silva, G. Cielniak, G. Wang, and J. Gao, "Deep learning-based crop row detection for infield navigation of agri-robots," *Journal of Field Robotics*, vol. n/a, no. n/a, 2023. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rob.22238

[12] D. Aghi, S. Cerrato, V. Mazzia, and M. Chiaberge, "Deep semantic segmentation at the edge for autonomous navigation in vineyard rows," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 3421–3428.

[13] C. W. Bac, J. Hemming, and E. J. Van Henten, "Stem localization of sweet-pepper plants using the support wire as a visual cue," *Computers and electronics in agriculture*, vol. 105, pp. 111–120, 2014.

[14] W. Wang, C. Li, K. Wang, L. Tang, P. F. Ndiluau, and Y. Cao, "Sugarcane stem node detection and localization for cutting using deep learning," *Frontiers in Plant Science*, vol. 13, p. 1089961, 2022.

[15] T. Kalampokas, E. Vrochidou, G. A. Papakostas, T. Pachidis, and V. G. Kaburlasos, "Grape stem detection using regression convolutional neural networks," *Computers and Electronics in Agriculture*, vol. 186, p. 106220, 2021.

[16] R. P. Haff, D. C. Slaughter, and E. Jackson, "X-ray based stem detection in an automatic tomato weeding system," *Applied engineering in agriculture*, vol. 27, no. 5, pp. 803–810, 2011.

[17] L. A. Wells and W. Chung, "Real-time computer vision for tree stem detection and tracking," *Forests*, vol. 14, no. 2, p. 267, 2023.

[18] W. b. James, "Greenhouse irrigation - what's the best watering system?" Jul 2023. [Online]. Available: https://greenhouseemporium.com/greenhouse-irrigation-systems/

[19] A. Benjumea, I. Teeti, F. Cuzzolin, and A. Bradley, "YOLO-Z: improving small object detection in yolov5 for autonomous vehicles," *CoRR*, vol. abs/2112.11798, 2021. [Online]. Available: https://arxiv.org/abs/2112.11798

[20] X. Fang *et al.*, "Design and implementation of constant pressure water supply monitoring system based on stm32," in *2017 IEEE 17th International Conference on Communication Technology (ICCT)*. IEEE, 2017, pp. 1487–1491.

[21] A. Rosebrock, "Object detection: Bounding box regression with keras, tensorflow, and deep learning," Jun 2023. [Online]. Available: https://shorturl.at/psPU2

[22] Z. Miao, K. M. Gaynor, J. Wang, Z. Liu, O. Muellerklein, M. S. Norouzzadeh, A. McInturff, R. C. Bowie, R. Nathan, S. X. Yu *et al.*, "A comparison of visual features used by humans and machines to classify wildlife," *BioRxiv*, p. 450189, 2018.

[23] M. Dominguez-Rodrigo, A. Fernandez-Jauregui, G. Cifuentes-Alcobendas, and E. Baquedano, "Use of generative adversarial networks (gan) for taphonomic image augmentation and model protocol for the deep learning analysis of bone surface modifications," *Applied Sciences*, vol. 11, no. 11, p. 5237, 2021.