Proceedings of the International Conference on Inventive Computing and Informatics (ICICI 2017)
IEEE Xplore Compliant - Part Number: CFP17L34-ART, ISBN: 978-1-5386-4031-9

Adding intelligence to the robotic coconut tree climber

Rajesh Kannan Megalingam, Rajesh Gangireddy, Gone Sriteja, Ashwin Kashyap, Apuroop Sai Ganesh
Department of Electronics and Communication Engineering, Amrita University, Amritapuri, India
rajeshkannan@ieee.org, rajeshrdy2511@gmail.com, sriteja.gone97@gmail.com, n.ashwin.k@ieee.org,apuroopkgs@gmail.com

Abstract— **Vision is one of the major component's of Automated Robotics and Artificial Intelligence. Camera's or Sensors are the one's which provide vision to robots. Only with help of this, a robotic arm is able to sense its environment and act accordingly to the environment. The information on geometrics from a vision device can be used to perform positioning or tracking tasks for the robotic arm. In this paper, we discuss about the method used in positioning and tracking tasks for the Robotic arm using the laser range method to find the distance from the camera to the target.**

Index Terms— FPV, Radian offset.

I. Introduction

With an ever-escalating demand for agricultural products and the decrease in the skilled labors for harvesting, agricultural robots come into the picture. "Farming is not preferred by younger generations," says Saverio Romeo, a principal analyst at Bath-based Beecham Research and also the co-author of the report "Towards Smart Farming: Agriculture Embracing the IoT Vision". Machines in agriculture played the crucial role in expanding the productivity and decreasing the cost of products used for the agricultural process. In this paper, we discuss the designing and implementation of a vision device on a coconut tree climbing robot. This aids in remote controlling and automation of the robotic arm. Real-time video transmission of the First Person View of the robotic arm to the user and the information on geometrics from a vision device are used in performing the tasks for tracking and positioning for the robotic arm. The camera system is able to identify the target in real time. The Laser Range method is used to locate the detected target as described in this paper. This system is able to find the distance to a single point on an object than to the entire object as such. This information can be used to guide the robotic arm to act on a single and definite point on the object. The further sections details on how a laser pointer and a single camera can be configured together to provide single machine vision to help obtain the distance information.

II. Problem Statement

As the robotic arm is being designed primarily for the coconut tree climbing robot, the target to be cut by the arm may not be clearly visible from the ground. The user may not be able to place the end effector (a circular saw blade in this case) properly for cutting. This problem is solved by having a camera system that can wirelessly stream the live video as a First Person View (FPV) of the robotic arm, to the user. The robotic arm is automated by solving inverse kinematics on ROS platform. Solving inverse kinematics require the object to be located in either tool coordinate system or base coordinate system. For the past few years, the advanced vision based and self-guided robotic arms for agricultural purposes have been available commercially. However, the situation is not the same in most of the developing countries such as India and other south Asian countries. Imparting technology into agriculture is still a concept on papers and research work in labs. Agriculture is an industry which is taken by low-income people. The amount of investment is fairly less in the agriculture sector which offers resistance to accept the high-cost technology into agricultural machines. A number of research works have been successful as projects but they failed as products for two reasons. One, the cost being too high, Two, the uneducated farmers would stick onto the conventional techniques as they see a factor of risk in investing money on top of learning new technology. In order to be practically usable, our system is optimised in terms of cost and ease of use. There are many distance finding methods available such as ultrasonic, infrared, and industrial laser rangefinders. However, weight is a primary concern when using such devices on the coconut tree climbing robot. Instead of adding new components, it is best to utilize the existing functionality of the components present on the robotic arm as much as possible. As there is a USB camera already being used for real-time video transmission, it is decided to use the same camera for distance calculation. Distance can also be determined by using two webcams that can provide stereo machine vision. The major drawback of this is that the second camera adds more weight and the complexity involved is also high.

III. Related works

In paper [1], the authors discuss and present the basic techniques involved in the machine vision system like image acquisition, image enhancement, segmentation etc., for agriculture robotics. This paper also weighs the pros and cons of the laser-equipped cameras. Whereas in the paper [2], the authors present and propose their idea of where a parametric trajectory is given which is called the visibility checking algorithm, it tells us if a 3D object remains in the field of view. The authors also state that their algorithm can be further expanded and integrated with to various path planning algorithms and mainly probabilistic roadmaps. The authors in the paper [4], the authors presented their idea on effective detection of fruits using various sensing systems. They have discussed on how hindrance, variable lighting conditions and clustering impact the detection of fruits and their solutions address the problem. Similar to the paper [4], authors in the paper [5] presented their idea of detecting fruits using deep neural networks. Not only in improving the accuracy, this method is also faster in deploying the fruits, as this does not require pixel-level annotation but requires the bounding box annotation. In paper [6] the authors show multiple sensor systems used to proficiently distinguish, localize, map and track the fruits in a commercial plantation. A numerous perspective approach is utilized to take care of the issue of impediment, hence keeping away from the requirement for work escalated field calibration to assess real yield. The fruits are distinguished in pictures using an R-CNN detector which is faster and best in the class, pair-wise correspondences are set up between pictures utilizing direction information given by a routing framework. The authors in the paper [7], discussed the different difficulties faced and the successes of sensing systems in order to detect the fruits. The authors also checked on the capacities of sensors utilized for localization of fruits. Challenges include the impediment, bunching, and varying light conditions. This paper also reviews the innovative machine-vision frameworks for localization of fruit and its recognition for automated collecting as well as yield load estimation of claim to tree crops which includes apples, pears, and citrus. Similarly, the authors of the paper [8], discussed the view of the delegate vision techniques required for robots used in harvesting purpose. The authors evaluated on the hand and eye coordination schemes and their different applications in robots. In paper [9], the authors proposed a system for tracking the pedestrians over a wide area, also in an open area such as railway stations, city centers, exhibitions using an array of laser range scanners. A simulation on the computer is conducted to examine the system performance with respect to different crowd density situations. In paper [10], a distance estimation technique is implemented using a single camera by approximating it to be a pinhole camera. This method uses triangular similarity and the results are also quite promising. However, there two major drawbacks of this method. The one- this method requires the width of the object (in either inches or cm) as an input to the system. Second - this method estimates the distance to the entire object as such and not on a specific point on it. In paper [11], it discusses the method for detecting the pedestrians from a real-time video and the braking action can be performed once detected. Blob analysis is performed on the foreground to detect the pedestrian, the control action and alert is responsible for the braking performed using a controller and actuator. Paper [12] discusses technique that can help detect the number of people in different scenarios, environmental conditions, and camera orientations. Circular Hough transforms, frame differencing and the histogram of oriented gradient-based methods are used for this.

IV. SYSTEM ARCHITECTURE

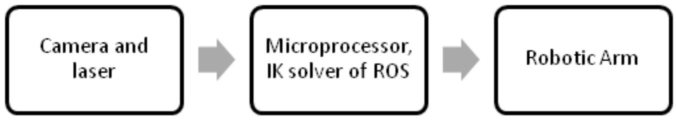

Fig 1. Block diagram of the entire system

The video transmission system is implemented as shown in figure 2. The camera is mounted on a base plate which has pan and tilt mechanism. This base is fixed on an actuating mechanism. The camera is attached to a raspberry pi 2 (model B) via a USB cable. This raspberry pi has an Wi-Fi module (2.4GHz 802.11b/g/n 150Mbps) on another port as shown in figure 3.The raspberry pi is made to automatically to a Wi-Fi Access point (WPA-secured network) while booting up and simultaneously a MJPEG Stream script (appendix) is executed. The live video can be accessed on devices that are connected to the same network as raspberry pis at http://IP:8080/?action=stream, where IP is the Internet Protocol address of the Raspberry PI. The video stream at 5 FPS had almost negligible latency and decent quality at 640x480 frame size.

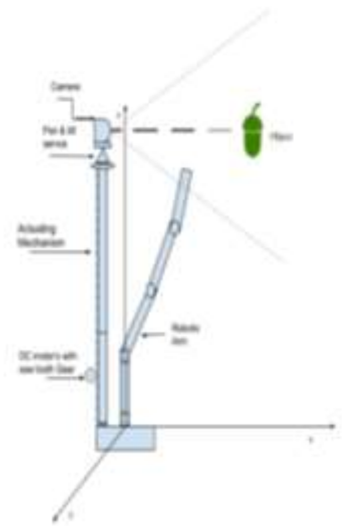

Fig. 2. Video streamer setup

The robotic arm needs the position (x,y,z coordinates) of the object to act on the target. These values are obtained using the camera and the laser and these are given to a microprocessor unit interfaced with a inverse kinematic solver of ROS and then these values are passed to the robotic arm to reach the target as shown in figure 1. For obtaining ,inverse kinematics solutions, the input required are the location of the object in terms of (x,y,z) coordinates with respect to the base of the arm.

Fig 3. Schematic diagram of video streamer setup

The robotic arm and camera setup are parallel to each other as shown in above figure. This means that the y and z coordinates of the camera and the end effector are same. Only x coordinate value changes for the arm and the camera. This implies that once the distance to the object is calculated, the x coordinate value for the arm can be calculated by subtracting the distance with the gap between the bases of arm and pole carrying the camera setup. The camera is actuated till it faces the object in a line of sight. This tuning is done by the user by monitoring the live video. The z coordinate can be calculated using the number of turns the DC motor turned in order to actuate the set up to that point. The y-coordinate can be calculated using the z-axis and the angle made by the object in the image. As shown in figure, if DC is the distance measured between the camera and the object, and is the angle made by the object point to be cut or accessed, with the camera, then the y-coordinate DY can be calculated using DY = DC *tan(α), as shown in figure 4. Finding distance to the object is the major task of finding the position of the arm.

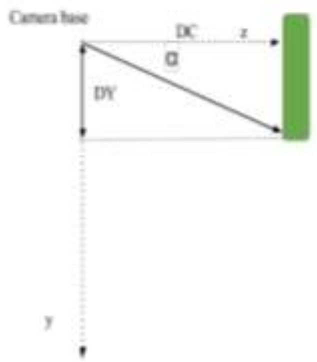

Fig. 4. Top view, to calculate the y coordinate from z coordinate

A beam of the laser is projected onto an object in the view field of the camera which can help to find the distance between the object and the camera. This beam of the laser is ideally parallel to the camera's optical axis. The laser dot is captured with the rest of the frame by the camera. Figure 5 shows the camera and laser trigonometry setup. A basic algorithm is run on the image which looks for the bright red (color of the laser) pixels. Assuming that the laser is the brightest area of the contour, the on the image, the laser dots position can be found in the image. Then for calculating the distance to the object based on where the horizontal axis of the image the laser dot falls. The closer the dot is to the center of the image, the farther away is the object. The entire process of finding coordinates as described above is shown as a flowchart in figure 6.

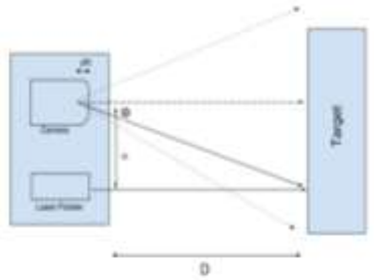

Fig 5. Camera and laser trigonometry.

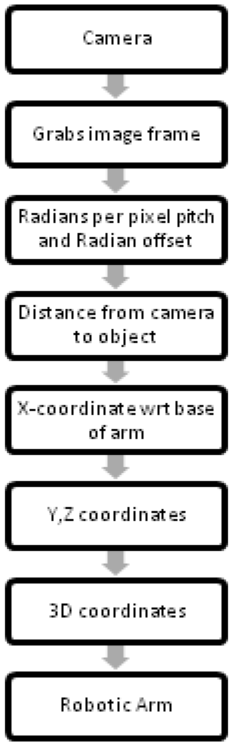

Fig 6. Flowchart for finding the coordinates.

V. Implementation

A beam of the laser is projected onto an object in the view field of the camera which can help to find the distance between the object and the camera. This beam of the laser is ideally parallel to the camera's optical axis. The laser dot is captured with the rest of the frame by the camera. Figure 7 shows the hardware setup of the laser range finder. The USB camera is powered by the USB of the Raspberry Pi. The laser operates at 3V consuming a maximum of 250mA producing a bright red laser beam of 650nm.The laser is powered from the main circuit of the coconut tree climbing robot via a voltage regulator. The pan and tilt mechanism is used to adjust and position the camera setup such that the laser pointer is pointed to the object to which the distance is to be calculated. The assumption that the laser dot in the image is the brightest red color, holds true for our case as the image is mostly filled with green color. Here, Distance is given by D =H/tan (ϴ); 'H' is the distance between the laser pointer and the camera.. is the angle between the red point pointed by laser and the perpendicular line of camera as shown in the figure. The angle 'ϴ' is calculated as pfc*rpc+ro Where 'pfc' is the number of the pixels from the center of the image in the frame captured, 'rpc' is the radians per pixel in the image and 'ro' is the radian offset used to compensate for the errors. In the equation given above, to calculate we will need to calculate 'pfc','rpc' and 'ro'. Only 'pfc' is known here. To get the values of radians per pixel and radian offset , a calibration is done. The object is kept at a range of known distances and the values are calculated for each distance using $=\tan^{-1}(H/D)$. A graph with pfc vs is plotted. The points, if measurements are taken correctly, will result in a straight line. The equation of any line can be represented by y=mx+c. Now, the values of radians per pixel and radian offset can be found by comparing the line equation with the equation =pfc*rpc+ro. While implementing this, we calibrated the system by measuring at 16 different distances.The entire process is implemented by using python with OpenCV and numpy libraries.

Fig 7. Hardware setup of the laser range finder

VI. RESULTS

Figure 8:- Images captured from the video stream from the camera setup attached to the robotic arm of the coconut tree climber.

The live video from the camera setup (shown in figure 8) had negligible lag at streaming speed of 5 frames per second, with 640x480 frame size.

Distance(cm)	pfc(pixels)	Theta(radians)
150	0	0.0765169843
145	3	0.0791446788
140	6	0.0819588499
135	9	0.0849800285
130	11	0.0882318656
125	14	0.0917417476
120	18	0.0955415608
110	26	0.1041670477
100	34	0.1144970268
90	46	0.1270890957
80	60	0.142771945
70	79	0.1628311845
60	108	0.18937005
50	139	0.226068388
40	185	0.2799498225
30	266	0.3660565158

Table 1. Theta for various distances.

Here, we have got two variables available, pixels from the center of the image(pfc) can be calculated and the 'ϴ' is calculated using $\tan^{-1}(H/D)$. By plotting these two variables on a graph as shown in figure 9, the two constants radians per pixel and radian offset can be obtained from the line equation.

By plotting the data shown in Table 1, and joining the points, we get the equation of connecting line as y=0.0010897179x+0.0760524984, indicating that rpc and ro are 0.0010897179 and 0.0760524984 respectively. If the connecting line is not a straight line, it indicates that the calibration data is not correct and that the system must be re-calibrated.

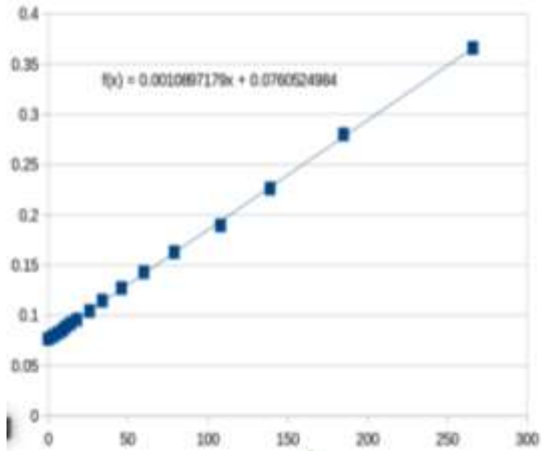

Fig 9. Calculating rpc and ro

Once the calibration is done, the system is tested by measuring distance to points kept at different scopes from camera. The results are shown in table 2.The results were extremely good, as the difference between the actual and calculated distance was not more than 0.50 cm for a range of distances.

Actual Distance-Da(In cm)	**Measured Distance Dc(In cm)**	**Error(Da-Dc) (In cm)**
15	15.02	-0.02
23	23.50	-0.50
25	25.00	0.00
36	36.10	-0.10
40	40.12	-0.12
45	44.76	0.24
50	49.69	0.31
55	55.10	-0.10
60	60.40	-0.4

Table 2. Results

VII. Conclusion

By implementing the ’Adding intelligence to the agricultural coconut tree climbing robot,’ robot used for coconut harvesting can be totally automated. The algorithm used for this process is simple and effective.

Acknowledgment

We wish to thank Almighty God who gave us the opportunity to successfully complete this venture. The authors wish to thank Amrita Vishwa Vidyapeetham University and the Humanitarian Technology (HuT) labs for aiding us in this endeavour.

References

[1] Nielsen, M.; Slaughter, D.; Gliever, C.; , "Vision-based 3D Peach Tree Reconstruction for Automated Blossom Thinning," Industrial Informatics, IEEE Transactions on , vol.PP, no.99, pp.1, 0

[2] Simon Leonard, Elizabeth A. Croft, James J. Little, "Dynamic Visibility Checking for Vision-Based Motion Planning," IEEE Int Conf. on Robotics and Automation, pp. 2283-2288, 2008.

[3] Hiroki Tanaka and Yuichi Tsumaki, "Development of a 6-DOF Camera Arm with Realistic Sensation," Proc. of the 23 d Annual Conf. of the Robotics Society of Japan, 3B23, 2005.

[4] Gongal, A. et al. “Sensors and systems for fruit detection and localization: A review.” Computers and Electronics in Agriculture 116 (2015): 8-19.

[5] DeepFruits: A Fruit Detection System Using Deep Neural Networks
Inkyu Sa * , Zongyuan Ge, Feras Dayoub, Ben Upcroft, Tristan Perez and Chris McCool.

[6] Hung, C.; Underwood, J.; Nieto, J.; Sukkarieh, S. A Feature Learning Based Approach for Automated Fruit Yield Estimation. In *Field and Service Robotics: Results of the 9th International Conference*; Mejias, L., Corke, P., Roberts, J., Eds.; Springer: Berlin, Germany, 2015; pp. 485–498.

[7] Annamalai, P., Lee, W.S., 2003. Citrus yield mapping system using machine vision. In: ASABE meeting presentation, Paper Number: 031002, St Joseph, MI.

[8] A. Arefi, A.M. Motlagh, K. Mollazade, R.F. Teimourlou, Recognition and localization of ripen tomato based on machine vision, Aust. J. Crop Sci., 5 (2011) 1144-1149.

[9] H. Zhao, R. Shibasaki, "A Novel System for Tracking Pedestrians Using Multiple Single-Row Laser-Range Scanners", *IEEE Trans. on Systems Man and Cybernetics - Part A: Systems and Humans*, vol. 35, no. 2, pp. 283-291, 2005.

[10] R. K. Megalingam, V. Shriram, B. Likhith, G. Rajesh and S. Ghanta, "Monocular distance estimation using pinhole camera approximation to avoid vehicle crash and back-over accidents," *2016 10th International Conference on Intelligent Systems and Control (ISCO)*, Coimbatore, 2016, pp. 1-5.

[11] *Sandeep, A.K., Nithin, S., Ramachandran, K.I*,"An Image processing based pedestrian detection system for drivers assistance," *2016 International journal of Control theory and Application*, 2016, pp. 7369-7375.

[12] *Raghavachari, C., Aparna, V., Chithira, S., Balasubramanian, V*.,"A comparative study of vision based human detection techniques in people counting applications," *2015 Procedia Computer science*,” 2015, pp. 461-469