Design of an intelligent motion control system for an agricultural field prowler using a path-following algorithm

Ach. Dafid[1*], *Faikul* Umam[2], *Hanifudin* Sukri[3], *Firmansyah* Adiputra[1], *Puteri* Nurul Ma'Rifah[3], and *Tsabitah* Tholi'atul Haq Al Faruq[3]

[1]Departement of Information System, Universitas Trunodjoyo Madura, Bangkalan, Indonesia
[2]Departement of Electrical Engineering, Universitas Trunodjoyo Madura, Bangkalan, Indonesia
[3]Departement of Mechatronics Engineering, Universitas Trunodjoyo Madura, Bangkalan, Indonesia

Abstract. The creation of robots that can move automatically and be used as a medium for simulating plowing movements is one of the efforts to improve efficiency in the soil cultivation process. Along with the development of robotics every year, line-following robots have become one of the most widely used technologies due to their ability to follow predetermined paths. In this study, a prototype path-following robot was created and designed to follow a path automatically, so that it could be used as a testing tool to simulate plowing movement patterns. This robot moves along a path by utilizing a camera as the main sensor to determine the robot's position in real time. The control method used is kinematic control, because this method allows for precise adjustment of forward speed and turning speed so that the robot remains on the predetermined path. Based on tests conducted on three different field sizes, the kinematic control system was able to provide good performance with an accuracy rate of 93.49% on a 2-square-meter field, 94.49% on a 1.5-square-meter field, and 95.49% on a 1-square-meter field. These results indicate that the kinematic control approach is effective in keeping the robot's movement stable and accurate when following a path automatically.

1 Introduction

1.1.Background

The The development of robotics science and technology is currently progressing rapidly and has entered the era of autonomous systems. Many companies and the general public have begun to develop and create technological devices equipped with artificial intelligence that are capable of performing various human tasks and reducing the risk of workplace accidents. In Indonesia, technological development has been seriously promoted by the government since 2010, one of which is through the allocation of a budget of 1.9 trillion rupiah for research and development in various sectors, including agriculture. The five main sectors in agriculture, namely plantations, forestry, food crops, fisheries, and livestock, have great potential to support national economic growth if managed optimally. To that end, the government is implementing an agricultural industry modernization program as an effort to increase productivity and encourage all elements of society to play an active role in a more efficient cultivation system [1].

One of the technologies widely used to support the automation process is robots. The term "robot" comes from the Czech word robota, which means worker. A robot generally consists of mechanical components, electronics, a processor-based control system, and control software [2]–[7]. Robots play an important role in replacing jobs that are repetitive, require high precision, or operate in hazardous environments. Fundamentally, robots are composed of three important parts: sensors as senses to detect the environment, a motion control system as an information processor like the human brain, and mechanical components as actuators to produce motion [8]–[11].

In agriculture, soil cultivation is the most energy- and time-consuming activity. This activity is generally carried out by plowing or leveling the soil using animal or machine power. In traditional methods, animals are used as traction power with the help of humans to control their path. Meanwhile, in semi-modern methods, animal power is replaced by machines, but the operator still has to control the direction of the machine's movement directly. This condition makes it difficult for farmers because the plowing process requires great physical strength and can pose a risk of injury if safety equipment is not provided [19][20].

To overcome this problem, this study proposes a path-following robot-based approach that can move automatically. This system utilizes a camera as a vision sensor to determine the robot's position on the track through wireless communication using Wi-Fi. Since this study focuses on the robot's ability to follow the path consistently, a kinematic control method is used to regulate the forward speed (linear velocity) and turning

* Corresponding author: ach.dafid@trunojoyo.ac.id

speed (angular velocity) so that the robot can maintain movement according to the predetermined trajectory [21][22].

The kinematic control method itself is an approach based on the principles of robot kinematics, a branch of physics that studies the relationship between position, orientation, velocity, and motion of objects without considering the forces that cause them. In the context of moving robots, kinematics is used to determine the relationship between wheel rotation, orientation direction, and robot displacement in two-dimensional space. This method is closely related to path following algorithms because it ensures that the robot moves stably on a predetermined path.

The purpose of this research is to improve the quality of agricultural land management through automation technology. The developed path following robot system is expected to support the effectiveness and productivity of land cultivation, especially for corn farmers [23][24]25]. The application of this technology has the potential to reduce the time, energy, and cost of land cultivation and increase farmers' interest in applying modern agricultural methods in the Madura region. The contribution of this research is to design a prototype path-following robot based on camera sensors and kinematic control methods, as well as to evaluate its performance on various field sizes to demonstrate the effectiveness of the system in following paths automatically and accurately.

2 Methodology

2.1 Kinematic control

Kinematic control is a method built on robot kinematics. From the kinematic equations, the relationship between the geometric concepts of the robot and the coordinates used to determine the position of an object. With the kinematics model, input reference configurations can be determined that must be fed to each actuator so that the robot can move to reach the desired position [12]–[18]. The following is an example of the kinematics of the robot in Fig. 1.

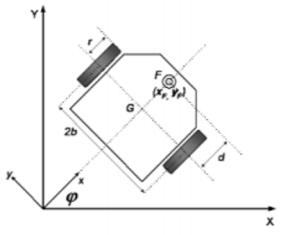


Fig. 1. Robot at XY coordinates.

The robot is assumed to be in cartesian XY coordinates. The parameters in Fig. 1 are:

1. θ is the facing angle of the robot
2. 2b is the width of the robot measured from the center of wheel 1 to the center of wheel 2
3. R is the radius of the wheel
4. d is the distance between the midpoints between the 2 wheels, point G and reference point F

To find the value of the distance traveled by the right and left wheels in meters, you can look for it using Equation (1) which can then be simplified into Equation (2) to find the distance for the right wheel, while to find the distance for the left wheel, you can use Equation (3), by dividing the number of rotary pulses measured on the robot by the number of rotary pulses in one rotation of the wheel, the value of the rotation of the wheel will be obtained, then multiplied by the circumference of the wheel on the robot. The distance traveled by the robot can be found by Equation (1)[5].

$$\text{Wheel circumference} \cdot \frac{s_r = \text{the right rotary pulse is measured}}{\text{the right rotary pulse is measured}} \quad (1)$$

$$2\,\pi\, r \cdot \frac{s_r = \text{the right rotary pulse is measured}}{\text{the number of rotational pulses per wheel revolution}} \quad (2)$$

$$2\,\pi\, r \cdot \frac{s_1 = \text{measured left rotary pulse}}{\text{the number of rotational pulses per wheel revolution}} \quad (3)$$

$$s = \frac{s_r + sl}{2} \quad (4)$$

Then referring to the applicable control law, the general speed equation can be stated in Equation (5) below [7].

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} Cos\,\theta & 0 \\ Sin\,\theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} \quad (5)$$

Where :
υ = linear speed value
ω = angular velocity value
θ = the value of the robot direction angle
a = the distance between the two wheels and the center point of the robot.
ẋd = desired robot speed value at position x
ẏd = desired robot speed value at position y

Inverse kinematics is a method that inputs a destination point and can generate an angle to achieve that goal. The following is the inverse kinematic 2 DOF mathematical equation in Equations 6 and 7.

$$\theta 1 = Atan2(Px, Py) \quad (6)$$

$$2 = Salty\,(\frac{P2 - L1}{L2}) \quad (7)$$

The following is a 3 DOF inverse kinematic mathematical equation in equation 8, equation 9, equation 10, equation 11, and 12.

$$r = \pm\sqrt{(Px^2 + Py^2} \quad (8)$$

$$D = \pm\sqrt{((Pz - L1)^2 - r^2} \quad (9)$$

$$\theta 1 = Atan2(Px, Py) \quad (10)$$

$$\theta 2 = Acos(\frac{D^2 - L2^2 - L3^2}{2L2L3}) \quad (11)$$

$$\theta 2 = Atan2(r, Px, Py) - Atan2(L2 + L3 * Cos\theta 3, L3 * Sin\theta 3) \quad (12)$$

Where:
Px = Position X
Py = Y position
Px = Z Position
θ1 = Angle 1
θ2 = Angle 2
θ3 = Angle 3
L1 = Arm Length 2
L2 = Arm Length 2
L3 = Arm Length 3

Forward kinematics is a method that inputs a joint configuration (pose) that produces movement from degrees of freedom. The following is the forward kinematic equation 2 DOF in equations 13, 14 and 15 [7].

$$Px = Cos\,(\theta 1) * \big(L2 * Cos\,(\theta 2)\big) \tag{13}$$

$$Py = Sin\,(\theta) * (L2 * Cos\,(\theta 2)) \tag{14}$$

$$Pz = L1 + L2 * Sin(\theta 2) \tag{15}$$

The following is the forward kinematic equation 3 DOF in equations 16, 17 and 18.

$$Px = Cos(\theta 1) * (L2 * Cos(\theta 2) + L3 * Cos(\theta 2 + \theta 3)) \tag{16}$$

$$Py = Sin(\theta 1) * (L2 * Cos(\theta 2) + L3 * Cos(\theta 2 + \theta 3)) \tag{17}$$

$$Pz = L1 + L2 * Sin\,(\theta 2) + L3 * Sin(\theta 2 + \theta 3) \tag{18}$$

Where:
Px = Position X
Py = Y position
Px = Z Position
θ1 = Angle 1
θ2 = Angle 2
θ3 = Angle 3
L1 = Arm Length 2
L2 = Arm Length 2
L3 = Arm Length 3

2.2 Accuracy level and error rate

The level of accuracy is carried out to determine the precision of a measurement system, namely measuring the closeness of the measurement to the original or actual value. While % error is done to find out the level of error, namely by comparing the results of the read measurement value with the original value of the various measurements you want to make. To calculate the accuracy value, it is necessary to calculate the % error first. Calculating the error value has several ways or methods that are carried out to measure it including using the Root Mean Square Error (RMSE) method as in Equation (19), Mean Square Error (MSE) as in Equation (20), Mean Absolute Error (MAE) and continued from MAE namely Mean Precentage Absolute Error (MAPE) as in Equation (21) and Equation (22) and Accuracy as in Equation (23).

$$\text{RMSE} = \sqrt{\sum \frac{(\text{Mark } Set\ Point - \text{True Value})^2}{\text{Lots of Data}}} \tag{19}$$

$$\text{MSE} = \sum \frac{(\text{Mark } Set\ Point - \text{True Value})^2}{\text{Lots of Data}} \tag{20}$$

$$\text{MAE} = \sum \frac{(\text{Mark } Set\ Point - \text{True Value})^2}{\text{Lots of Data}} \tag{21}$$

$$\text{MAPE} = \sum_{t=1}^{n} \left| \frac{\text{True Value} - \text{Mark } Set\ Point}{\text{True Value}} \right| \text{X } 100\% \tag{22}$$

$$\text{Accuracy} = 100\% - Error \tag{23}$$

This research made a robot that can follow a path that can be used for plowing the fields. The robot is able to follow the trajectory that has been made with input from the camera connected to the laptop and the image processing results are sent to the robot with wifi communication. Then the robot will walk according to the trajectory that has been made.

2.3 Mechanical design

In general, the design planning of this tool is made aiming to find out the information and placement of the arrangement of supporting components such as sensors and actuators of the track follower robot. In Figure 2 and Fig. 2 are the results of the design of the tool and the trajectory follower robot.

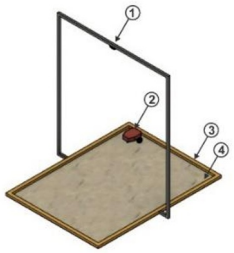


Fig 2. Overall display of tools and robots.

Figure 2 is the result of the overall track follower robot tool design with a height on both poles of 2.75 cm. For a description of the numbering and function of the track follower robot as follows:

1. Camera
   The camera which functions as a device for reading the position of the robot and the ground, is then processed on a laptop and forwarded to the robot. The camera used is a Logitech c920 wabcam camera with a recording speed of 720p.
2. Robots
   The robot functions as a tool or test medium for the system to be made.
3. Land Limits
   The Land Boundary functions as a differentiating indicator between the land that the robot will pass from and the land that the robot will not pass through.
4. Land

The land functions as a media that will be passed by the robot in carrying out its duties according to the commands sent by the laptop. The land used is square with a maximum size of 2 square meters and a minimum of 1 square meter.

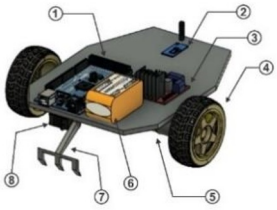


Fig 3. Track follower robot design.

Fig. 3 is the result of the design of the path follower robot design. For a description of the numbering and function of the track follower robot as follows:

1. ESP32
2. L298 DC Motor Drivers
3. DC motor wheel
4. DC motors
5. Lipo Battery
6. Hijacking Tool
7. Servo Motors

2.4 System architecture and block diagram

This control system block diagram is used to find out the work system or process of how the system works. Block diagram of the path follower robot movement control system to follow the path using the kinematic control method as shown in Figure 4.

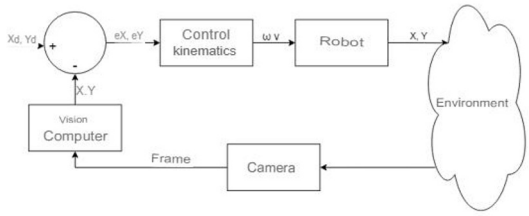


Fig. 4. System block diagram.

The control system block diagram in this study is to map the processes in the system from input to the expected output with the closed loop control system used. The block diagram of the path-following robotic plowing robot movement control system for path tracking using the Kinematic Control method can be explained as follows:

1. Input to the system is in the form of coordinates in real time on land which is denoted by the variables Xd, Yd and robot coordinates which are denoted by the variable X as the length of the coordinates of the x position, Y as the length of the coordinates of the y position. Input the value of Xd as the set point coordinates at position X, Yd as the set point coordinates at position Y, Xr and Yr are coordinates that the camera reads from the robot at positions X and Y. The set point coordinates are determined based on the coordinates of the land and the robot detected by the camera.
2. The sensor used in this control system is a camera. Where the capture results from the camera are processed to produce readings in the form of coordinate values for the robot and the land.
3. The Kinematic Control Method functions as a controller that will determine the direction of movement of the rice field plowing robot following the path to get to the desired position. Xd, Yd, X and Y are the inputs of this research method. While the output of the Kinematic Control method is ω to set the turning speed and υ for the forward speed to the destination point.
4. The path follower robot in the system is an actuator from the kinematic method system.
5. The environmental block is the media or field that will be traversed by the robot.
6. The computer vision block is the process of translating camera readings into X and Y coordinates.
7. The output of this system is the coordinates of the robot's position.

2.5 System workflow and flowcharts

Figure 5 is a general system flowchart. This process is the overall process of the path follower robot system. In the early stages, the thing that is done is the land detection process until the robot has finished following the path. This process is carried out in a relatively short time.

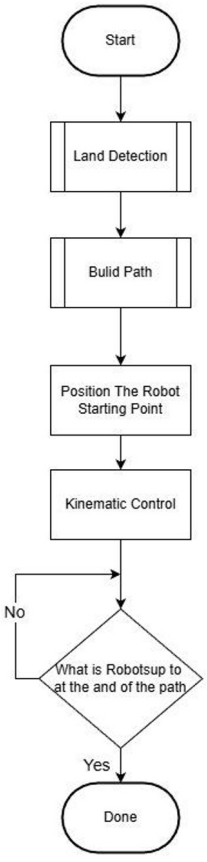


Fig. 5. System general flowchart.

The first step in this flowchart system is the land detection process using a camera device as a sensor connected to a laptop. Furthermore, the results of camera readings are processed to create path points and connect each path that is made. Then the camera detects the robot and sends commands for the robot to position itself at the starting point of the path and follow the coordinates of the path that has been made to the robot system using wifi transmission communication. If the robot's position detected by the camera does not match the path made, the process on the laptop sends an order to return to the path made using the kinematic control method by turning the bow according to the robot's position to the position according to the path. If the camera detects that the robot has arrived at the last path that was made, the robot will stop and finish.

1. Land Detection Flowchart

Fig. 6 is a land detection flowchart. This process is the process of detecting the length and width of the land. The initial thing to do at this stage is the process of capturing images to get a value on the variable.

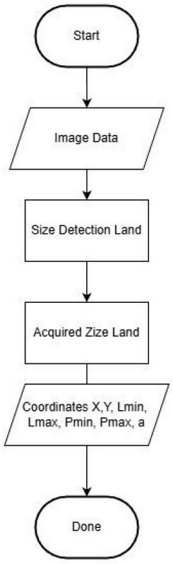


Fig. 6. Land detection flowchart.

The first step in this flowchart system is image capture, which is the process of reading image data obtained from the results of camera detection, then the land detection process. When land can be detected, land size is obtained. Furthermore, the land size data is processed to take the X, Y coordinates of the land width. Then the X and Y coordinates of the land are stored in the Pmax, Pmin, Lmax and Lmin variables, namely variables for storing the minimum and maximum length width.

2. Flowchart Building Path

In Fig. 7 is a flowchart of building paths. This process is the process of creating and compiling paths on the camera system. The first thing to do at this stage is to make a path point algorithm on the land by calculating using the length and width of the land obtained from the camera readings and then storing them in the X and Y variables.

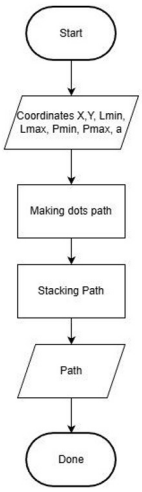


Fig. 7. Flowchart building path.

The first step is to make the path building algorithm, namely at the first X coordinate point determined by means of the minimum land width plus the width of the path distance, for the first Y coordinate point it is determined by means of the minimum land length plus the width of the path distance. The second X coordinate point is determined by means of the minimum land width plus the path spacing width, for the second Y coordinate point it is determined by the maximum land length minus the path spacing width. At the third X coordinate point it is determined by means of the maximum land width minus the path distance width, for the third Y coordinate point it is determined by the maximum land length minus the path distance width. The fourth X coordinate point is determined by means of the maximum land width minus the path distance width, for the fourth Y coordinate point is determined by the minimum land length plus the path distance width.

Calculations with this formula are repeated until the length or width of the field cannot be divided again from the width of the robot. Then the results of each calculation are connected sequentially to the last as shown in Figure and the following is the path point determination formula:

$$\mathrm{E\,to-i} = (\mathrm{X} = \mathrm{Lmin} + \mathrm{a}, \mathrm{Y} = \mathrm{Pmin} + \mathrm{a}) \quad (24)$$

$$\mathrm{F\,to-i} = (\mathrm{X} = \mathrm{Lmin} + \mathrm{a}, \mathrm{Y} = \mathrm{Pmax} - \mathrm{a}) \quad (25)$$

$$\mathrm{G\,to-i} = (\mathrm{X} = \mathrm{Lmax} - \mathrm{a}, \mathrm{Y} = \mathrm{Pmax} - \mathrm{a}) \quad (26)$$

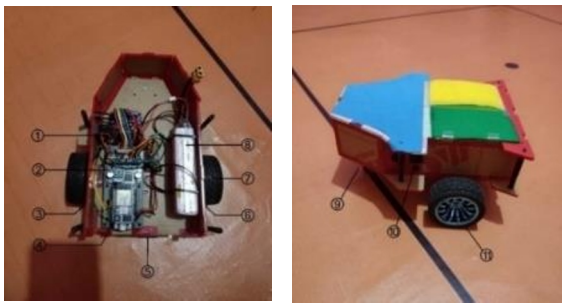
$$\mathrm{H\,to-i} = (\mathrm{X} = \mathrm{Lmax} - \mathrm{a}, \mathrm{Y} = \mathrm{Pmin} + \mathrm{a} \quad (27)$$

Where:
Pmin = minimum length of land
Pmax = maximum length of land
Lmin = minimum width of land
Lmax = maximum width of the land
X = point X coordinates
Y = Y point coordinates
a = distance between desired paths

3 Result and discussion

3.1 Camera calibration and object detection results

Making a path follower robot must go through several design stages which include mechanical design, electronics design and programming. Thus, the results obtained are path follower robots that function as robots to help facilitate human work. In the following, the results of the overall path follower robot design will be presented. This path following robot is made of acrylic which has a box dimension of 20 cm x 16 cm x 5 cm (without wheels) and has an average thickness of 3 mm. For overall dimensions with wheels 20 cm x 23 cm x 10 cm as shown in Fig. 12.



(a) (b)

Fig. 8. (a) Mechanical results and (b) Path follower robot components.

In Fig. 8 (a) is the view from the inside of the robot, (b) is the view from the outside of the robot. In the appearance of the robot there are several components that are used to support the running of the robot which consists of a battery, microcontroller, and cables as a link between components. On the outside of the robot, there are several components that support the robot's running, including the DC motor wheel, the gearbox, the on/off switch button located on the left side of the robot. For numbering information can be described in the explanation below.

Description of the number in Figure 8 is as follows:

1. Motor Driver
2. Step Down
3. Left Motor
4. Microcontroller
5. Servo
6. Motor Right
7. Right Wheel
8. Battery
9. Free wheel
10. Switch/ON OFF Button
11. Left Wheel

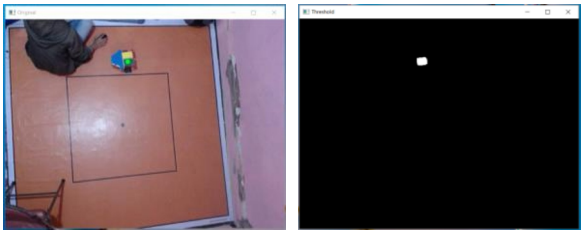
3.1.1 Green color

In this stage the treatment of the camera to be able to detect green color is calibrated to get the green HSV value on the object. So the minimum H value = 40, the maximum H = 101, the minimum S = 0, the maximum S = 255, the minimum V = 0, and the maximum V = 255.

Table 1. Green camera testing.

Time	Mark					
	H-	H+	S-	S+	V-	V+
Afternoon	50	100	70	255	0	200
Evening	50	91	70	255	0	193

In table 1 above is a table of the range of HSV values used in green. The value used is the value between the minimum and maximum in each variable from the HSV. The HSV value used for green is not that difficult because the green color and the base color of the field are quite different.



(a) (b)

Fig. 9. (a) Original camera and (b) Green threshold of camera.

Fig. 9 a and b above are the results of green color recognition calibration which will be used to detect green color when running the system. In Figure 9 (a) is a designation of the HSV value used in detecting color. In Fig.9 (b) is the result of visual processing from Fig. 9 (a) in the camera reading process, in this process all colors other than the introduced color will be black, but otherwise the introduced color will be white.

The green color calibration results show that color segmentation has high stability because the contrast between green and the base color of the field is quite clear. This finding is consistent with the research by Ramadhani et al. [2], which states that the HSV color space is more stable to lighting changes than RGB. This is important because path follower robots require consistent path detection for optimal kinematic control.

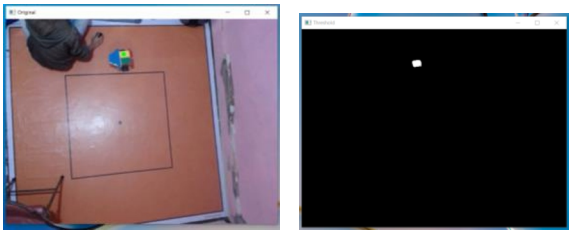
3.1.2 Yellow color

In this stage, the camera is treated to be able to detect yellow, a calibration is carried out to get the yellow HSV value on the object. Then the value of minimum H = 21, maximum H = 75, minimum S = 0, maximum S = 255, minimum V = 151, and maximum V = 255.

Table 2. Yellow camera testing.

Time	Mark					
	H-	H+	S-	S+	V-	V+
Afternoon	0	55	80	255	95	255
Evening	0	50	85	255	97	255

Table 2 above is a table of the range of HSV values used in yellow. The value used is the same as before, namely the value between the minimum and maximum for each variable from the HSV. The HSV value used for yellow is the most difficult because the yellow color is not very much different from the base color of the field, and it will be quite difficult when the base color used is white.



(a) (b)

Fig. 10. (a) Original Camera and (b) Threshold Yellow Color Camera.

Figure 10 a and b above are the results of the yellow recognition calibration which will be used to detect yellow when running the system. In Figure 10 (a) is a designation of the HSV value used in detecting color. In Figure 10 (b) is the result of visual processing from Figure 10 (a) in the camera reading process, in this process there are only 2 colors displayed, namely black and white.

Yellow color calibration has a higher level of difficulty due to the Value (V) being sensitive to light. This condition is in accordance with the findings of Liu et al. [7], which explains that colors with high intensity tend to produce noise in HSV-based detection. However, the right threshold setting allows the system to reliably detect yellow colors on the robot path follower test track.

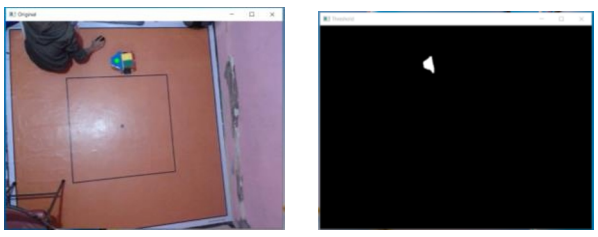
3.1.3 Blue color

In this stage the treatment of the camera is the same as for green and yellow to be able to detect blue, a calibration is carried out to get the blue HSV value on the object. Then the value of minimum H = 92, maximum H = 114, minimum S = 165, maximum S = 255, minimum V = 0, and maximum V = 255.

Table 3. Testing the blue camera.

Time	Mark					
	H-	H+	S-	S+	V-	V+
Afternoon	90	120	100	255	47	195
Evening	91	120	100	255	47	193

Table 3 above is a table of the range of HSV values used in blue. The value used is the value between the minimum and maximum in each variable from the HSV. The HSV value used for the blue color is not that difficult because the blue color and the base color of the field are quite different.



(a) (b)

Fig. 11. (a) Original Camera and (b) Threshold Blue Color Camera.

Figure 11 a and b above are the results of the blue color recognition calibration which will be used to detect the blue color when running the system. In Figure 11 (a) is a designation of the HSV value used in detecting color. In Figure 11 (b) is the result of visual processing from Figure 11 (a) in the process of reading the camera, in this process all colors other than the introduced color will be black, but otherwise the introduced color will be white.

Blue color detection provides the most stable results due to its high contrast with the color of the ground and surface area. A study by Fan et al. [9] also shows that blue is often used as a marker because it is consistent despite variations in light. Therefore, blue is effective for use at important points along a path follower robot's route, such as the destination area or path boundary markers.

3.1.4 Camera sensor detection test

The path-following rice field plowing robot uses a camera as a sensor with the Logitech C920 HD type. This test is carried out to determine the position of the robot in the field which is a comparison with the position of the original robot. The following is a reading view of the Command Prompt as shown in Fig. 12.

```
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
Send to Client:  b'430,422,425,110,1,15,'
```

Fig. 12. Camera sensor reading results.

Testing the camera sensor functions to read the position of the robot in the field so that the system can give the next command. The camera position at the time of testing can be seen in Figure 13.

Fig. 13. Camera sensor testing.

Table 4. Testing the reading of robot coordinates on the camera against the field.

No	Camera Sensors (Centimeter)	Measuring Tool (Centimeter)	Accuracy Level	% Error
1	25	25	100 %	0 %
2	50	48	96 %	4 %
3	75	70	93,3 %	6,7 %
4	100	92	92 %	8 %
AVERAGE			95,325 %	4,675 %

In table 4 above are the results of a comparison of tests from camera readings with direct measurements using manual measuring tools. In the comparison of these tests there is a difference between the camera sensor readings and the measuring instruments used manually. This test was carried out 4 times with different sizes in each test. From this difference, the difference value or error value is obtained from the camera reading with its original state, the highest error value is in the 4th test, which is 8% in the test with a size of 100cm in its original state. Test results show that the camera has an average position reading accuracy of 95.3%, with an error of 4.7%. This value is in line with the research by Dewi et al. [4], which reported an accuracy of 92–96% in visual tracking robots. The difference between camera readings and manual measurements is influenced by the camera's viewing angle, robot distance, lens distortion, and lighting conditions. The main finding of this section is that the camera is accurate enough to support the kinematic control of the path follower robot, as the system only requires consistent position estimates. Limitations at this stage include sensitivity to lighting, decreased accuracy at greater distances, and perspective distortion due to the use of a single camera. The implication is that improved lighting, camera position adjustments, or the use of more than one camera could improve position identification accuracy in further research.

3.2 Robot mechanical and movement test results

At this stage is the process of forming a path or point of interest in the movement of the path-following robot. The reason for using a spiral shape in the testing process is due to get differences with other studies conducted by other people and to make it easier to make paths [16]. In this test the examiner uses a field size of 2 square meters. The following in Figure 14 below is the first path point of the 14 corner points.

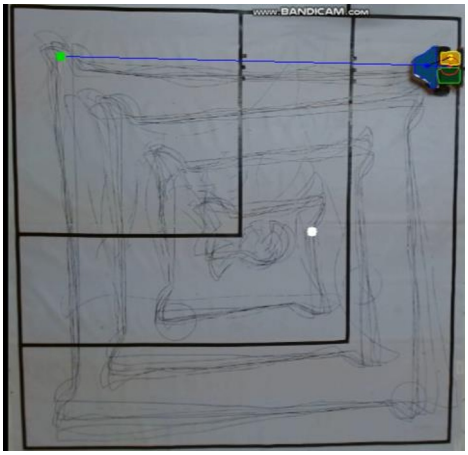


Fig. 14. The first path point on a 2 meter square field.

3.3 Field testing (2m × 2m, 1.5m × 1.5m, 1m × 1m)

Testing of the path following robot is carried out on the land surface which is carried out on a square field with a length of 2 meters and a width of 2 meters or the same as the camera reading which is 480 x 480. Testing on a rectangular field measuring 2 meters wide and 2 meters long as shown in Fig. 15.

Fig. 15. Field testing 2 meters x 2 meters.

The results of the robot movement on a field size of 2 meters long and 2 meters wide with the Kinematic Control method can be seen in Figure 16 below.

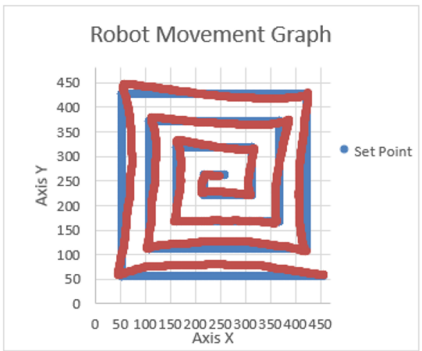


Fig. 16. 2-Meter field robot movement graph.

The results of the experiments carried out obtained data as much as 1635 data on the coordinates of the robot's position read by the camera, the data is attached in Appendix 2. The robot starts from the coordinates of the robot x 459, y 57 on the camera and ends at coordinates x 250 and y 257 on the camera. From these data, the RMSE, MSE, MAE and MAPE values were

obtained using Equation 6 to find the RMSE value, Equation 7 to obtain the MSE value and to obtain the MAE and MAPE values using Equation 8 and Equation 9. From these equations each obtained a value of 10, 64245 for RMSE, 113.26177 for MSE value, for MAE value is 8.62202 while 6.51134 for MAPE value. The largest RMSE value in this 2m × 2m field indicates that a larger area causes an increase in visual noise and more significant changes in robot orientation. This finding is consistent with the research by Liu et al. [7], which states that kinematic control accuracy decreases as the navigation area increases. Lighting variations in larger fields also cause errors to increase.

Robot Road Test Field Size 1.5m x 1.5m this test is carried out to obtain an error value based on Equation 6 to Equation 9 so that it can determine the success rate of the robot. The purpose of this test is to measure the performance of robot movement using the Kinematics Control method. Testing on a square field measuring 1.5 meters wide and 1.5 meters long as shown in Figure 17.

Fig. 17. Field testing 1.5 meters x 1.5 meters.

The results of the robot movement on a field size of 1.5 meters long and 1.5 meters wide with the Kinematic Control method can be seen in Figure 18 below.

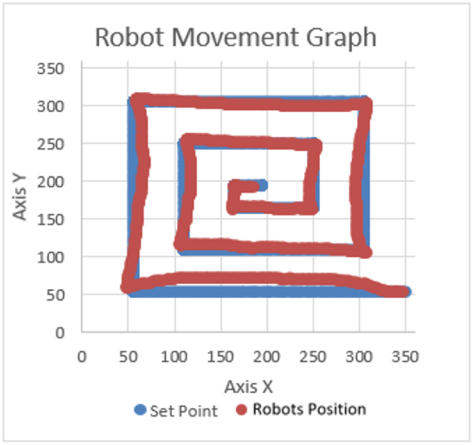


Fig. 18. Graph of field robot movement 1.5 meters.

In the experiments conducted in this field, we obtained 944 data coordinates for the position of the robot read by the camera, the data is attached in Appendix 3. The robot starts at coordinates x 347 and y 54 on camera readings and ends at coordinates x 185 and y 193 on camera reading. From these data, the RMSE, MSE, MAE and MAPE values are obtained using the same method as in Equation 6 to find RMSE values, Equation 7 to obtain MSE values and to obtain MAE and MAPE values using Equation 8 and Equation 9. From these equations, each -each value is 7.17635 for RMSE, 51.5 for MSE value, for MAE value is 5.2839 while 5.50427 for MAPE value. The decrease in RMSE in the 1.5m × 1.5m field shows that kinematic control works more stably in medium-sized areas. This value is close to the results of Guo et al.'s research [8], which recorded an RMSE of around 7 in small autonomous vehicles. This shows that a more consistent camera distance and a not-too-large area provide optimal conditions for robot direction correction.

Robot Road Test Field Size 1m x 1m. The next path-following rice field plowing robot test is carried out on a square field with a length of 1 meter and a width of 1 meter or the same as the camera reading, which is 240 x 240. This test is carried out the same as the previous test, namely obtaining error values based on Equation 6, Equation 7, Equation 8 and Equation 9 so that it can determine the success rate of the robot. The purpose of this test is to measure the performance of the robot's movement using the Kinematics Control Test method on a square field measuring 1 meter wide and 1 meter long as shown in Figure 19.

Fig. 19. Field Testing 1 meter x 1 meter.

This test obtained the results of robot movement on a field size of 1 meter long and 1 meter wide with the Kinematic Control method. This experiment starts from coordinates x 231 and y 54 and ends at coordinates x 135 and y 147 based on camera readings. This figure can be seen in Figure 20 below.

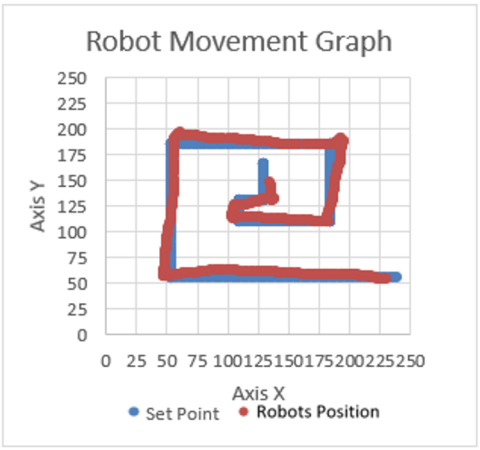


Fig. 20. Graph of field robot movement 1.5 meters.

In this experiment, 391 robot position data were obtained, robot position coordinate data detected by the camera, the data is attached in Appendix 4. From these data, the RMSE, MSE, MAE and MAPE values were

obtained using the same method as in Equation 6 to find the value RMSE, Equation 7 to get the MSE value and to get the MAE and MAPE values using Equation 8 and Equation 9. From these equations it is obtained that each value is 5.19615 for RMSE, 27 for MSE value, for MAE value is 4.40665 while 4 .50105 for the MAPE value. The smallest RMSE value in the 1m × 1m field indicates that small areas provide the best visual stability. This is because the camera's viewing angle does not change much, so the robot orientation correction process (ω) is faster and more precise. These results support the findings of Fan et al. [9], that narrow navigation spaces produce smaller errors in mobile robots. This shows that kinematic control is very effective in limited areas.

3.4 Performance evaluation (RMSE, MSE, MAE, MAPE)

Experiments that have been carried out in 3 different fields, the overall results obtained can be seen in Table 5.

Table 5. Results of robot movement data in 3 different fields.

Field	Error Calculation Method			
	RMSE	MSE	MAE	MAPE
2 meters	10,64245	113,26177	8,62202	6,51134 %
1,5 meters	7,17635	51,5	5,28390	5,50427 %
1 meters	5,19615	27	4,40665	4,50105 %

Based on Table 5, it can be concluded that the experiments conducted on 3 different fields had an RMSE error value of no more than 11, for errors using the MSE method the highest was 113.26177 for a field measuring 2 square meters, as well as using the MAE method. the highest error does not touch the value of 10 only to the value of 8.62202 and using the MAPE method the lowest error percentage is below 10%, namely 6.5% and it can be concluded that the wider the field used, the greater the error value obtained will be.

4 Conclusion and suggestions

This work demonstrates that the incorporation of a single-camera vision system with kinematic control effectively stabilizes the movement of a path-following robot. The robot reliably adheres to the designated trajectory and exhibits enhanced precision as the area diminishes. This underscores that visual-based control is appropriate for small to medium field applications.

This study's theoretical contribution is the demonstration of combining differential-drive kinematic models with real-time visual coordinate extraction to produce linear and angular velocity commands for autonomous robots. This offers novel insights on employing computer vision as the exclusive feedback mechanism in kinematic control frameworks. The study's shortcomings are: (1) susceptibility to lighting conditions, (2) diminished accuracy with larger field sizes, (3) perspective distortion from a single above camera, and (4) studies conducted solely in controlled indoor environments.

Future research should use additional cameras or supplementary sensors like IMUs to enhance position estimation, evaluate the system in actual outside agricultural environments, and investigate more sophisticated control techniques such as Model Predictive Control or adaptive control. Integration of machine learning-based visual detection may enhance resilience across diverse environmental circumstances. This study enhances the understanding of vision-based autonomous robotics by presenting actual evidence about the impact of field size, color segmentation precision, and camera stability on robotic motion performance. These discoveries are anticipated to facilitate the advancement of more dependable, cost-effective agricultural robots in the future.

Based on the results of the research that has been done, the conclusions are obtained:

1. Camera sensor calibration or configuration greatly affects the reading and detection of robots or objects.
2. Camera sensor readings affect the response of the robot's movement to do its job.
3. The use of the type of camera and the quality of the camera used greatly influences the results of camera readings.
4. The choice of color for the detection material greatly influences the ease of calibration
5. From several tests, the best result is the PWM value for the robot speed, which is 140 PWM.
6. Based on the results of tests carried out on 3 different fields, the RMSE error value is not more than 11. For errors using the MSE method, the highest value is 113.26177 for a field measuring 2 square meters, as well as using the MAE method with the highest error value. not touching the value 10 only up to the value 8.62202 and using the MAPE method the lowest error percentage is below 10% and it can be concluded that the wider the field used, the greater the error value obtained will be.

References

1. A. W. Utomo, Transisi agraris ke industri (studi sosiologis perubahan sosial: transisi masyarakat agraris ke industri di Dusun Timang, Wonokerto, Kabupaten Wonogiri). Jurnal Cakrawala. **7**, 2, 205–230 (2018)
2. R. A. Ramadhani, G. Jati, W. Jatmiko, A. Y. Husodo, Adaptive multi-strategy observation of kernelized correlation filter for visual object tracking. Proc. Asia-Pacific Conf. Intell. Robot Syst. (ACIRS). 134–139 (2019). https://doi.org/10.1109/ACIRS.2019.8936042
3. F. Guangyuan, W. Ming, H. Shuai, F. Wenyu, W. Xiangyao, Object tracking algorithm based on better parallax of monocular under unknown environment. Proc. Int. Comput. Signals Syst. Conf. (ICOMSSC). 830–834 (2018). https://doi.org/10.1109/ICOMSSC45026.2018.8941967

4. T. S. Dewi, R. Mardiati, L. Kamelia, A. E. Setiawan, Yumna, R. Rustandi, Prototype of mobile robot tracking object using sensor vision. Proc. Int. Conf. Wireless Telematics (ICWT). 1–5 (2022). https://doi.org/10.1109/ICWT55831.2022.9935442
5. C. Chen, et al., A real-time motion detection and object tracking framework for future robot-rat interaction. Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS). 7404–7409 (2021). https://doi.org/10.1109/IROS51168.2021.9636403
6. D. Han, Y. Peng, Human-following of mobile robots based on object tracking and depth vision. Proc. Int. Conf. Mechatronics Robotics Autom. (ICMRA). 105–109 (2020). https://doi.org/10.1109/ICMRA51221.2020.9398366
7. Y. Liu, Y. Zheng, L. Han, J. Liu, Z. Pan, F. Sun, The moving target recognition and tracking using RGB-D data with the mobile robot. Proc. Chinese Control Conf. (CCC). 4342–4347 (2019). https://doi.org/10.23919/ChiCC.2019.8865493
8. N. Guo, X. Zhang, Y. Zou, B. Lenzo, T. Zhang, A computationally efficient path-following control strategy of autonomous electric vehicles with yaw motion stabilization. IEEE Trans. Transp. Electrification. **6**, 2, 728–739 (2020). https://doi.org/10.1109/TTE.2020.2993862
9. S. Lai, M. Lan, B. M. Chen, Model predictive local motion planning with boundary state constrained primitives. IEEE Robot Autom Lett. **4**, 4, 3577–3584 (2019). https://doi.org/10.1109/LRA.2019.2928255
10. Y. Chen, B. Chu, C. T. Freeman, Iterative learning control for robotic path following with trial-varying motion profiles. IEEE/ASME Trans. Mechatronics. **27**, 6, 4697–4706 (2022). https://doi.org/10.1109/TMECH.2022.3164101
11. M. Latif, S. Hardi, S. Herawati, Motion control development for autonomous ground robots in agriculture task. J Phys Conf Ser. 2193(1), 012060 (2022). https://doi.org/10.1088/1742-6596/2193/1/012060
12. M. K. Mishra, A. K. Samantaray, G. Chakraborty, Joint-space kinematic control of a bionic continuum manipulator in real-time by using hybrid approach. IEEE Access. **10**, 47031–47050 (2022). https://doi.org/10.1109/ACCESS.2022.3171236
13. A. Singletary, S. Kolathaya, A. D. Ames, Safety-critical kinematic control of robotic systems. IEEE Control Syst Lett. **6**, 139–144 (2022). https://doi.org/10.1109/LCSYS.2021.3050609
14. N. Tan, P. Yu, Z. Zhong, F. Ni, A new noise-tolerant dual-neural-network scheme for robust kinematic control of robotic arms with unknown models. IEEE/CAA J Automatica Sinica. **9**, 10, 1778–1791 (2022). https://doi.org/10.1109/JAS.2022.105869
15. M. M. Marinho, B. V. Adorno, Adaptive constrained kinematic control using partial or complete task-space measurements. IEEE Trans. Robot. **38**, 6, 3498–3513 (2022). https://doi.org/10.1109/TRO.2022.3181047
16. L. Tang, F. Yan, B. Zou, K. Wang, C. Lv, An improved kinematic model predictive control for high-speed path tracking of autonomous vehicles. IEEE Access. **8**, 51400–51413 (2020). https://doi.org/10.1109/ACCESS.2020.2980188
17. J. Fan, L. Jin, Z. Xie, S. Li, Y. Zheng, Data-driven motion-force control scheme for redundant manipulators: A kinematic perspective. IEEE Trans. Ind. Inform. **18**, 8, 5338–5347 (2022). https://doi.org/10.1109/TII.2021.3125449
18. R. Kazemian, S. Nahavandi, R. D. Saddik, Vision-based tracking control for mobile robot navigation. IEEE Access. **9**, 17782–17793 (2021). https://doi.org/10.1109/ACCESS.2021.3054843
19. M. Zhu, K. Hauser, Real-time camera-based robot navigation using visual odometry. Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS). 11465–11472 (2020). https://doi.org/10.1109/IROS45743.2020.9341270
20. A. S. F. M. Ashraf, M. S. Uddin, J. Shin, Path-following control for autonomous mobile robots using kinematic modeling. IEEE Access. **8**, 94448–94460 (2020). https://doi.org/10.1109/ACCESS.2020.2995979
21. S. Kamble, A. Raina, Autonomous mobile robot navigation using deep visual perception. Proc. IEEE Int. Conf. Robot. Autom. (ICRA). 4450–4456 (2021). https://doi.org/10.1109/ICRA48506.2021.9560983
22. C. L. Teo, B. M. Chen, Vision-guided autonomous mobile robot navigation using image-based visual servoing. Proc. IEEE Int. Conf. Robot. Autom. (ICRA). 1022–1028 (2021). https://doi.org/10.1109/ICRA48506.2021.9561117
23. A. H. Abdulhafez, S. Kamel, Real-time vision-based detection and tracking for mobile robot navigation. IEEE Access. **9**, 11892–11905 (2021). https://doi.org/10.1109/ACCESS.2021.3050864
24. M. S. Mahmoud, H. Abouelnaga, Visual servoing-based mobile robot motion tracking under uncertain environments. IEEE Access. **10**, 58472–58484 (2022). https://doi.org/10.1109/ACCESS.2022.3181574