J@WIM
International Journal of Wireless and Multimedia Communications
Vol. 3 No. 1 (2026), January 2026, pp. 12-19
ISSN: 3047-2172

Implementation Of Legged Robot Control With Pixy Camera Module Through Image Processing Technique

Riza Widia[a*], Akbar Abadi[b*], Satria Jaya[c], Alfhadila[d]

[abcd]*Electrical Department, Politeknik Negeri Padang, 25166, Indonesia*

[d*] *fhadilfhadil27313@gmail.com*

***Abstract* —This tool is the development and implementation of a robot control system using a Pixy camera module through image processing techniques to support automatic navigation. This system is designed to improve the robot's navigation capabilities in a dynamic environment by utilizing image processing as the main method in detecting and recognizing objects and obstacles around it.The purpose of applying the Pixy camera to SAR robot control is to make the robot able to distinguish between victims and other objects, controlling robot movements with the Pixy camera so that the robot can move to pick up victims, knowing how the Pixy camera works in detecting victims in SAR robot navigation, knowing how to calibrate and process Pixy camera images for automatic navigation on SAR robots.The method used in this research is the integration of Pixy camera module with Arduino Mega microcontroller to process image data through serial communication. Image processing techniques are applied to identify the optimal object and search for the object continuously. The Pixy camera functions as the main sensor to detect colored objects that have been programmed, and the data obtained is used by the microcontroller to direct the movement of the robot. The test results show that the robot is able to navigate independently with a high level of accuracy in various environmental scenarios. This implementation shows significant potential in the development of autonomous robots that are more intelligent and adaptive to changes in the surrounding environment.**

Keywords: Legged Robot; Pixy Camera Module; navigation; Image processing

Manuscript received 29 Sep. 2024; revised 2 Oct. 2024; accepted 15 Oct. 2024. Date of publication 4 Feb. 2026.

I. INTRODUCTION

The development of technology and industrial automation is increasingly rapid, sophisticated and modern.and modern encourage humans to fulfill their needs.So that robot technology was developed to help and facilitate the work of humans in the future. In addition to this, there is also a balance system which is an example of a feedback control system. Robotchave been developed by several countries around the world. Among them humanoid robots, wheeled robots, and mobile robots. Another example is a legged robot which is used as a competition in the Indonesian SAR Robot Contest (KRSRI), where the legged robot is used as a prototype robot for the Indonesian SAR Robot Contest (KRSRI). where a legged robot is used as a prototype robot to rescue victims from danger.

design, manufacture and programming of robots accompanied by elements that emphasize search and rescue missions. The contest emphasizes the search and rescue mission of disasters that commonly occur in Indonesia.This competition emphasizes search and rescue missions for disasters that commonly occur, especially in Indonesia.The theme for 2024, just like the theme for 2023 is still inspired by the difficulties that occur after earthquake disasters,such as the recent one in Cianjur in late November 2022.KRSRI 2024 is focused on navigating through various obstacles to illustrate the,The robot is also challenged to rescue victims from a specific location to a predetermined safe zone.rescue victims from a certain location to a predetermined safe zone. The theme for KRSRI 2024 is “Autonomous Robot for Post-Disaster Victim Rescue”.

Various obstacle conditions require participants to equip their robots with control systems and intelligence in processing their various sensors with various trajectory constraints.with control systems and intelligence in processing its various sensors with various trajectory constraints.various trajectory constraints.The legged autonomous robot at KRSRI 2024 is indeed,indeed cannot be directly applied to post-disaster locations, but with the existence of the KRSRI is expected to increase the potential of students' abilities in creating autonomous robots.in creating post-disaster legged autonomous robots. So that if later it will be developed a robot with a similar structure that is more realistic in size,it will be easier and faster to realize.

The obstacles that the robot will go through are four types of obstacles and save potential victims. Four types of obstacles in the form of sloping roads, broken roads, debris roads, and muddy roads as illustrations of real conditions after disasters, especially earthquakes. To reach the finish, the robot must walk down and up the stairs. Where the robot must be able to move on its own, look for victims on its own and make its own decisions without being controlled by a human operator [1].

In the previous year the SAR (Searc And Rescue) robot from the Minangkabau team of Politeknik Negeri Padang did not use a camera in navigating victims, in the absence of a camera there were obstacles, namely the robot was difficult to find victims. victims, when robots search for victims with other types of sensors such as ultrasonic, there are often errors and failures to take victims. there are often errors and failures to take victims. From the problem the use of ultrasonic as a sensor to find victims, a sensor change in the form of a Pixy camera.

Pixy cameras were chosen in the implementation of the SAR robot where the main reason for using them is that they are easier, more accurate and more efficient in victim navigation than ultrasonic sensors. It is easy to use,the circuit is not complicated and available in the market. In addition, this Pixy camera is flexible because its readings can determine an object based on the color captured by the camera and can send block information to the u color captured by the camera and can send block information to the Arduino at a speed of 1 Mbits/second, which means Pixy can process 60 frames per second [2].

The SAR robot can move when the switch is turned on, this robot moves through obstacles and looks for victims by detecting obstacles, victims and compass angle values. Using an ultrasonic distance sensor to read walls, a CMPS-03 compass sensor so that the robot does not deviate whenwalking and a Pixy camera for robot navigation in rescuing victims, all data from these sensors is processed by Arduino Mega.data from the sensor is processed by arduino mega. OpenCM 9.04 as a microcontroller controlling the robot's legs and arduino mega as a microcontroller controlling the robot that gives instructions.controller that provides automatic instructions that have been programmed based on the sensor readings used.

II. METHOD AND MATERIALS

A. METHOD

The method used is the image processing method on the Pixy cam where when the robot is run using a switch to start searching for victims the robot will track with a predetermined set point value.

Utilizing Image processing as the main method in detecting and recognizing objects and obstacles around it. The Pixy camera module is used as the main visual sensor integrated with the robot control system.

B. MATERIALS

1. Pixy Cam

Pixy Cam is a smart camera that uses color recognition algorithms to detect and track objects in various applications, such as robotics, games, and automation projects. It can recognize objects by color and provide information about their position and size. Pixy Cam also comes with a video output feature that allows users to see what the camera sees. This is useful for calibration adjustments on the Pixy cam to the object or color you want to record[3].

Fig 1:Pixy Cam

2. Arduino Mega 250

Arduino is a microcontroller-based board or open source electronic circuit board in which the main component is a microcontroller chip unit with the AVR type from the Atmel company. The microcontroller itself is a chip or IC (integrated circuit) that can be programmed using a computer. The purpose of embedding a program in a microcontroller is so that the electronic circuit can read input, process the input and then produce the desired output. So the microcontroller serves as the brain that controls the input and output process of an electronic circuit[4].

The Arduino Mega can be powered via a USB connection or with an External power supply. The power

Fig. 1 Arduino mega 250

source is selected automatically. External (non-USB) power can come from either an AC-DC adapter or a battery. The adapter can be connected by plugging its 2.1 mm center-positive plug into the electrical connecting board. The leads from the battery can be inserted into the GND and Vin pin headers of the power connector[5].

3. Open CM9.04

Open CM9.04 is a robot controller that is open source, both in hardware and software. In hardware, this controller is supported by an ARM Cortex-M3 microcontroller.

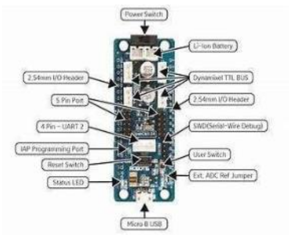

Fig. 2 Open CM9.04

It has additional features in Open CM9.04, namely 5V output for external modules, LED indicators for power status, communication, and other work indicators. For programming and development, it is supported by OpenCM IDE software (arduino-based), C/C++ programming language, and uses a DFU (Device Firmware Update) based bootloader. The application of Open CM9.04 is commonly used in robot control, and robotics applications[6].

4. Servo Motor

The servomotor is the structural unit of the servo system and is used with the servo drive. The servomotor includes a motor that drives the load and a position detection component. The servo system varies a controlled quantity, such as position, speed, or torque, according to a set target value (command value) to precisely control the operation of the machine[7]. There are 3 types of servo motors used in this SAR robot according to the needs, capabilities, and size, namely:

a. Servo Dynamixel

Dynamixel itself is a smart servo line-up from a well-known robot manufacturer from South Korea, ROBOTIS. ROBOTIS is also the developer and manufacturer of several popular robots such as TurtleBoot 3, ROBOTIS-OP, ROBOTIS DREEAM, OLLO, Bioloid, and so on. Dyynamixel is widely used by various companies, universities, and hobbies due to its versatile expansion capabilities and provides various functions and feedback data regarding position, voltage, speed, temperature, and so on. To access these features, a software called RoboPlus has also been provided, with this software you can perform servo hardware testing, change the servo ID, troubleshooting and so on[8]. Dynamixel can be used in various robots with multi-joint systems, such as arm robots, humanoid robots, snake robots, hexapod robots, and so on. The Minangkabau SAR team robot uses a dynamixel AX- 18A servo.

b. Servo MG996R

The MG996R servo is a high-torque servo widely used in daily projects and robotics applications that require greater power, with a stall torque of up to 11 kg-cm and a rotation angle of 180° [9]. This servo is used in SAR robots as a gripper to retrieve victims.

Fig. 3 Servo MG99R

c. Servo MG90S

The MG90S servo is a micro servo with a small size and metal gears, commonly used for applications that require actuation in confined spaces or small robotics projects. This servo has a stall torque of 2.2 kg-cm and a fast rotational speed, and the use of this servo in SAR robots as a gripper to grab the victim[10]

5. Pixymon Software

Pixymon is an application that allows you to configure Pixy2 and see what it sees. The app runs on several different platforms including Windows, MacOS and Linux. With Pixymon, users can monitor the output of the Pixy camera live, perform calibration for color and object recognition, as well as set various parameters such as the image capture zone and the type of filter applied [11].

6. Arduino IDE

Arduino IDE is software that is used to create programming sketches in other words Arduino IDE as a medium for programming on the board that you want to program [12]. Arduino IDE is made from JAVA programming language, which is equipped with C / C ++ (wiring) library, which makes input / output operations easier.

7. Robotics Software

Robotis software is used to sketch the SAR robot programming as a medium for programming on the board that you want to program [13].

8. Li-Po 3s Battery

Li-po or lithium polymer is a liquid battery. This type uses a solid electrolyte and can deliver power faster. This battery has a similar structure to Li-ion batteries, but is made of gel-like material (Silicon-Graphene) which is quite light in weight. Li-po batteries use polymers as their electrolyte, while Li-ion batteries use ion-based electrolytes in solid form. So that Li-po batteries are much more environmentally friendly than Li-ion batteries [14].

9. UBEC (Universal Battery Elimination Circuit)

UBEC is an electronic component that functions to reduce DC voltage to 5V or 6V voltage level, and the maximum input voltage depends on the UBEC specifications [15].

Fig. 4 UBEC (Universal Battery Elimination Circuit)

III. RESULT AND DISCUSSION

After the system has been developed, testing needs to be conducted. The testing is carried out to determine whether the device works well as planned. The purpose of the testing is also to identify the strengths and weaknesses of the device that has been created.

3.1 Robot Movement Testing

This test is conducted to determine whether the robot's movement execution aligns with the desired commands in the created program. The testing was conducted by calling the robot motion data on the opencm 9.04 microcontroller via parallel communication with the Arduino Mega. In this test, we aim to evaluate whether the robot's movement can be controlled as desired. Here are the results of the tests obtained based on the table below.

TABLE 1

ROBOT MOVEMENT TESTING

No.	Robot Summoning Data	Robot Movement
1.	Send data (1)	Foot Meeting Stop Position
2.	Send data (2)	Forward
3.	Send data (3)	Backward
4.	Send data (4)	Right Sideways
5.	Send data (5)	Left Sideways
6.	Send data (6)	Turn Right
7.	Send data (7)	Turn Left
8.	Send data (8)	Waist Forward
9.	Send data (9)	Waist Back
10.	Send data (10)	Stopping Position

Based on the robot movement testing table, it can be seen that the robot's movement data varies, where the robot moves according to the logic of the calls inputted into the program. Where the robot's movement data is adjusted according to the input from the Pixy camera sensor.

3.2 Measurement and Reading of Pixy Cam on Objects

The measurements taken by the Pixy camera include dominant color, Y coordinates, X coordinates, and the location within the camera's field of view. The efficiency and reliability of the Pixy camera readings are very important, especially in ensuring that the data produced is accurate for use in subsequent processes. Errors in reading and measurement can affect the overall performance of the robotic system that uses this camera as a sensor. In measuring and reading objects with the Pixy camera, the objects need to be calibrated so that the Pixy camera can read and identify them. Below are the measurement results in reading objects obtained from the Pixy camera based on the table below.

TABLE 2

OBJECT MEASUREMENTS AND READING

No.	Camera Capture Object Location	Object Color	Coordinates Y	Coordinates X
1	Center of camera capture plane	Orange	96	161 – 163
2	To the left of the camera capture plane	Orange	105 - 110	284 – 287
3	To the right of the camera capture plane	Orange	68 - 77	18 – 20
4	Above the camera capture field	Orange	168 - 173	161 – 164
5	Below the camera capture field	Orange	28 - 40	143 – 147
6	Unreadable object	None	-	-

Based on the table above regarding the Pixy camera, it can perform readings very efficiently and the measurement data results are accurate as desired for further processing. For the Pixy camera readings, the X coordinate range is from 0 to 319, while the Y coordinate range is from 0 to 199. The Pixy camera will not output the parameter value of a color if that color has not been stored by Pixy through the calibration process.

3.3 Testing the SAR Robot for Object Search

This test aims to evaluate the SAR robot's ability to locate objects in the search area. This process involves programming and search algorithms that enable the robot to scan and accurately recognize target objects. The testing determines where the target object is placed in the location for the robot to conduct the search. Parameters such as the number of objects successfully detected, measurement data of objects by the camera, and the detection error rate. The performance of robots in search tasks can be enhanced by utilizing data from previously analyzed measurements from the Pixy camera. The robot can recognize objects with the camera by storing the color data of the objects through a calibration process, which is carried out using the Pixymon software. Here are the results of the tests obtained based on the table below.

TABLE 3

TESTING THE ROBOT SEARCHING FOR OBJECTS

No.	Camera Detection	Object Color	Coordinates Y	Coordinates X	Robot Movement
1	Dolls	Orange	96	161 - 163	Robot stops and picks up object
2	Dolls	Orange	105 - 110	284 - 287	Robot moves left
3	Dolls	Orange	68 - 77	18 - 20	Robot moves right
4	Dolls	Orange	168 - 173	161 - 164	Forward motion robot

5	Dolls	Orange	28 - 40	143 - 147	Robot backward motion
6	None	-	-	-	Robot stop

Based on the table above, it can be seen that the movements of the robot being called vary, where the robot moves depending on the readings of the x and y coordinate values. The logic for calling the robot's movements is arranged according to the desired needs.

The image above is a program for testing a robot that searches for objects whose colors have been stored by the Pixy camera during the calibration process. This program includes a

Fig. 5 Testing When object on left

function to control the lamp status using Pixy.setLamp(1, 0); which means that 1 is the command to turn on the Pixy lamp, and an if statement that checks whether the number of blocks detected by Pixy.ccc.getBlocks() is greater than zero. There is also a for loop that reads the number of detected blocks, with the variables i, sig, x, and y used in this loop. Inside the loop, there is an assignment to read the object's signature, as well as the x and y coordinates from the array called Pixy.ccc.blocks[i].m_signature, Pixy.ccc.blocks[i].m_x, and Pixy.ccc.blocks[i].m_y. There is a print statement that functions to output the signature, x, and y coordinates to the serial interface.

Fig. 6 Testing When the Object is on the Right

At the beginning of the program, there is a switch-case conditional statement that checks the value of the variable (Value) and executes the corresponding block of code. Each case in the switch-case controls specific actions of the robot, such as moving the servo, reading camera input, and sending data via serial communication. In case 1, the robot will stop and pick up an object if certain conditions are met. This program also uses the servo.write() function to control the servo's movement, as well as the Serial.print() command to send data via serial communication. There are comments in the code, written in Indonesian, providing explanations about the actions being taken, such as moving forward, backward, turning, and stopping.

The image above displays the reading parameters of the x and y coordinates by the Pixy camera when the specified object is detected by the camera. The value of the coordinates changes depending on the position of the object in the camera's view. The object in the image above is located in the center of the camera's view with coordinate values of x=160 – 163 and y=95 – 96. When the object is in the center of the robot's camera

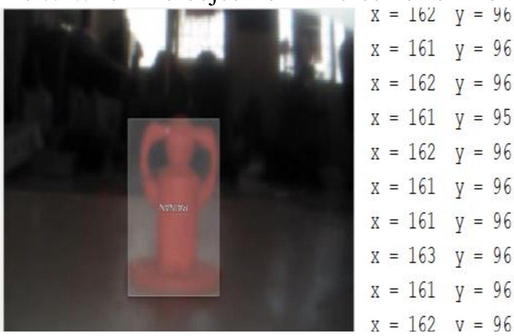

Fig. 7 Object on center

view, it will stop and pick up the specified object.

In the image resulting from the reading, the object on the left displays the x and y coordinate reading parameters by the Pixy camera when the specified object is detected. The object in the above image is located on the left side of the camera's view with x coordinate values of 285 – 287 and y coordinate values of 105 – 110. When the object is on the left side of the robot's camera view, it will move to the left until the object is in the center of the camera's view.

The image above displays the reading parameters of the x and y coordinates by the Pixy camera when the specified object is detected. The object in the image above is located on the right side of the camera view with x coordinates of 18 – 20 and y coordinates of 68 – 77. When the object is on the right side of the camera's view, the robot will move to the right until the object is in the center of the camera's view.

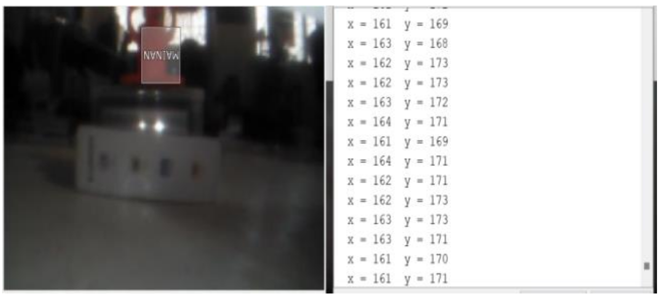

Fig. 8 Testing When the Object is Above

The image above displays the reading parameters of the x and y coordinates by the Pixy camera when the specified object is detected. The object in the image above is located at the center top of the camera view with x coordinates ranging from 161 to 164 and y coordinates from 168 to 173. When the object is above the camera's view, the robot will move forward until the object is centered in the camera's view. The movement of the robot depends on the specified y-coordinate parameters.

The image above displays the reading parameters of the x and y coordinates by the Pixy camera when the specified object is detected. The object in the image above is located at the bottom center of the camera's view with coordinate values of x=143 – 147 and y=28 – 40. When the object is below the camera's view, the robot will move backward until the object is centered in the camera's view, after which the robot will carry out other commands. The movement of the robot depends on the specified y-coordinate parameters.

In the image above, when the specified object is not detected by the Pixy camera, the Pixymon interface will not provide any indicators that something is in the camera's view.

Fig. 9 Testing When the Object is Under

Additionally, the Arduino serial monitor does not display any parameter values read by the camera when the object is not detected. When the specified object is not detected by the Pixy camera, the robot will remain stationary without doing anything, depending on the program that has been created

3.4 Testing SAR Robots for Object Differentiation

After identifying the objects, the next step is the robot's ability to distinguish between those objects. This test is to ensure that the robot can not only locate objects but also identify various types of objects, which is crucial in rescue missions where priority must be given to accurate object identification. The analysis includes the algorithm's ability to recognize differences between objects based on features measured by the Pixy cam, such as color, size, and shape. The reliability in distinguishing one object from another will be tested with several scenarios, including settings where the objects are similar to each other in several aspects. Here is the table of results from the robot's object differentiation test

TABEL 4

TESTING ROBOT DISTINGUISHING OBJECTS

NO	Camea Detection	object color	number of objects	Coordinates X,Y	Robot Movement
1	Dolls	Orange, Black, and Yellow	3	Orange : X = 148Y= 104 Yellow : X= 244Y= 104 Black: -	The robot moves to pick up the orange object
2	Dolls	Orange and Yello	3	Orange 1: X= 50Y= 89 Orange 2: X= 272Y= 108 Yellow: X= 170,Y= 93	Robot moves to pick up the nearest orange object
3	Dolls	Orange and Yellow	2	Orange: X= 240 Y=111 Yellow: X= 80Y= 96	Moving robot picks up orange object
4	Dolls	Orange	2	Orange 1: X= 224Y= 90 Orange 2: X= 61Y= 73	Moving robot picks up orange object

Based on the table above, it can be seen that the robot's movements are called according to the logic of the program created, where the robot differentiates objects based on the colors detected according to the program's logic. The program logic for the robot is that when a predetermined color is detected by the Pixy camera, the robot will approach or pick up the object that has been specified during the Pixy camera calibration process.

```
void loop(){
  switch(Nilai) {
  case 1 : {
  int i; pixy.setLamp(1, 0);
  pixy.ccc.getBlocks();
  if (pixy.ccc.numBlocks){
    for (i = 0; i < pixy.ccc.numBlocks; i++){
    int sig     = pixy.ccc.blocks[i].m_signature; Serial.print("Sig= "); Serial.print(sig);
    int x       = pixy.ccc.blocks[i].m_x;         Serial.print("  x= "); Serial.print  (x);
    int y       = pixy.ccc.blocks[i].m_y;         Serial.print("  y= "); Serial.println(y);
          if (sig == 1) {
          int rentang = map(x, 10, 250, 0, 50);
          servoX.write(rentang); delay(1000);
          Nilai = 2;}
        }
      }
   } break;
   case 2 : { servoY.write(0); delay (1000);
   servocapit.write(110); delay (1000);
   Nilai = 3; } break;
   case 3 : { servoY.write(160); delay(1000);
   servoX.write(30); delay(1000);
   servocapit.write(150); delay (1000);
   Nilai = 1; } break;
  }
}
```

Fig. 10 Robot Logic Program Differentiates Objects

The image above displays a program code written in C++ within an Integrated Development Environment. (IDE). The program includes the loop() function, which is a standard part of an Arduino program. Inside the loop() function, there is a switch-case control structure and if statements, which are used to execute different commands based on specific conditions. In case 1, the program checks the data block received from the sensor and moves the servo based on the detected value. If case 1 is completed, it proceeds to case 2 and so on according to the program; in cases 2 and 3, the program calls for the servo movements.

Fig. 11 Robot Distinguishes 3 Objects of Different Colors

The image above displays the reading parameters of the signature, the x and y coordinates by the Pixy camera when the specified object is detected. The objects in the camera's view are three in total: orange, yellow, and black. The colors detected by the Pixymon interface are only two: orange and yellow, as only those colors were saved during the calibration process. As for the black objects, no data storage is performed.

The object in the camera view displays signature data parameters with x and y coordinate values. The data for the orange object consists of sig=1, with x coordinate value=148 and y coordinate value=104. Meanwhile, for the yellow object, it consists of sig=2, with x coordinate value=244 and y coordinate value=104. In reading the object above, the robot moves according to the program, which is that the robot only moves to pick up orange objects.

Fig. 12 Robot Distinguishes 2 Same Color Objects 1 Different

The image above displays the reading parameters of the signature, the x and y coordinates by the Pixy camera when the specified object is detected. The objects in the camera's view are three in total: two orange and one yellow.

The object in the camera's view displays signature data parameters with x and y coordinate values. The data for the first orange object on the right side of the camera's view consists of sig=1, with x coordinate value=272 and y coordinate value=90, while the second orange object on the left side of the camera's view consists of sig=1, with x coordinate value=50 and y coordinate value=91. The yellow object consists of sig=2, with x coordinate value=174 and y coordinate value=92. In reading the objects above, the robot moves according to the program, which specifies that the robot only picks up orange objects. Since there are two orange objects, the robot takes them both alternately.

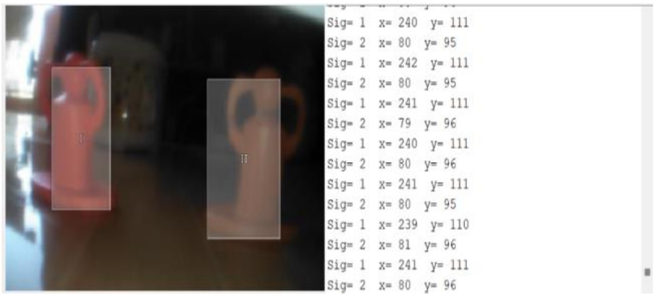

Fig. 13 Robot Distinguishes 2 Objects of Different Colors

The image above displays the reading parameters of the signature, the x and y coordinates by the Pixy camera when the specified object is detected. The objects in the camera's view are two: one is orange and the other is yellow.

The object in the camera view displays signature data parameters with x and y coordinate values. The data for the orange object consists of sig=1, with x coordinate value=241 and y coordinate value=111, while for the yellow object it consists of sig=2, with x coordinate value=80 and y coordinate value=96. In the reading of the object above, the robot only moves to pick up the orange object, even though the yellow object is detected in the parameters.

Fig. 14 Robot Distinguishes 2 Same Color Objects

The image above displays the reading parameters of the signature, the x and y coordinates by the Pixy camera when the specified object is detected. There are two orange objects in the camera's view.

The object in the camera view displays signature data parameters with x and y coordinate values. The data for the first orange object on the right side of the camera view consists of sig=1, with x coordinate value=62 and y coordinate value=73, while for the second orange object on the left side of the camera view, it consists of sig=1, with x coordinate value=224 and y coordinate value=90. In reading the objects above, the robot moves according to the program, which specifies that the robot only picks up orange objects. Since there are two orange objects, the robot takes them both alternately.

IV. CONCLUSION

1. After testing, it can be seen that the robot's movements vary, where the robot moves according to the logic entered in the program.
2. berdasarkan dari data percobaan,kamera pixy dapat melakukan pembacaan yang efisien dan hasil penggukuran yang sesuai, pembacaan kamera Pixy memiliki rentang koordinat X dari 0 hingga 319, sedangkan koordinat Y dari 0 hingga 199. The Pixy camera will not output the parameter value of a color if the color is is not stored by Pixy through the calibration process.
3. in testing the robot in searching for objects, it can be seen from the tests carried out that the robot moves depending on the reading of the x and y coordinate values.
4. The design and application of the Pixy camera on the SAR robot allows the robot to detect and recognize objects effectively. The camera is equipped with a built-in processor that allows real-time image processing.
5. The process of reading and transmitting data from the Pixy camera to the mega arduino involves serial communication which makes the transfer of image data efficient. The Arduino mega processes the data received from the Pixy camera to determine the navigation action of the SAR robot.
6. Image processing for automatic navigation on SAR robots involves analyzing images taken by Pixy cameras to efficiently determine colored objects. An image processing algorithm is used to direct the robot towards the victim.
7. The Pixy camera functions as a victim detection by recognizing the specific color or shape intended for the victim. The detection technology used by Pixy cameras is very helpful in improving the efficiency and success of SAR robot rescue missions.

References

[1] Panduan Kontes Robot Indonesia, buku 3 Kontes Robot SAR Indonesia (KRSRI) tahun 2024.(Online), (https://kontesrobotindonesia.id/kri-2024.html,diakses 20 Mei 2024).

[2] Hooking up *Pixy* to a Microcontroller (like an Arduino), tahun 2020. (Online) (https://docs.*Pixycam*.com/wiki/doku.php?id=wiki:v2:hooking_up_*Pixy*_to_a_microcontroller_-28like_an_arduino-29, diakses 20 Mei 2024).

[3] *PixyCam*, "*Introducing Pixy*2", tahun 2024. (Online) (https://*Pixycam*.com/, diakses 20 Mei 2024).

[4] S. Siswanto, M. Anif, D. N. Hayati, and Y. Yuhefizar, "Pengamanan Pintu Ruangan Menggunakan Arduino Mega 2560, MQ-2, DHT-11 Berbasis Android," J. RESTI (Rekayasa Sist. dan Teknol. Informasi), vol. 3, no. 1, pp. 66–72, 2019, doi: 10.29207/resti.v3i1.797.

[5] V. Alapján-, "Spesifikasi Arduino Mega 2560," vol. Arduino Me, pp. 1–23, 2016.

[6] Robotis, "*OpenCM IDE*", tahun 2024. (Online) (https://emanual.robotis.com/docs/en/software/opencm_ide/getting_started/, diakses 24 Juni 2024).

[7] Omron. "*Technical Explanation for Servomotors and Servo Drives",* tahun 2024.(Online)(https://www.ia.omron.com/support/guide/14/introduction.html, diakses 17 Juli 2024).

[8] DigiWare. "Panduan memilih servo dynamixel-digiware", tahun 2020. (Online)(https://digiwarestore.com/id/digiware-news/63_panduan-memilih servodynamixel, diakses 15 Juli 2024).

[9] Tower Pro, "MG996R high torque metal gear servo", tahun 2014. (Online) (https://www.towerpro.com.tw/product/mg996r/, diakses 17 Juli 2024).

[10] Tower Pro, "MG90S micro servo", tahun 2014. (Online)(https://www.towerpro.com.tw/product/mg90s-3/, diakses 17 Juli 2024).

[11] PixyCam, "Pixymon overview", tahun 2018. (Online)(https://docs.Pixycam.com/wiki/doku.php?id=wiki:v3:Pixymon_overview, diakses 17 Juli 2024).

[12] Erinta, "Mengenal perangkat Arduino IDE", tahun 2021. (Online), (https://www.kmtech.id/post/mengenal-perangkat-lunak-arduino-ide, diakses17 Juli 2024).

[13] C. N. Thai, "Exploring robotics with robotis systems," Explor. Robot. With Robot. Syst., no. 2009, pp. 1–342, 2017, doi: 10.1007/978-3-319-59831-4.

[14] Ini perbedaan antara baterai Li-on dengan Li-po- Brilio.net, tahun 2019. (Online) (https://www.brilio.net/creator/ini-perbedaan-antara-baterai-li-ion- dengan-li-po-4135ab.html, diakses 15 Juli 2024).

[15] Mengenal UBEC-NPCLab. tahun 2015, (0nline) (http://npclab.blogspot.com/ 2015/03/mengenal-ubec.html, diakses 15 Juli 2024).