

DEVELOPMENT OF AN AUTONOMOUS TENNIS BALL RETRIEVER ROBOT

Duru SARICALAR, Department of Mechatronics Engineering, Istanbul Technical University, Turkey, saricalar18@itu.edu.tr

([ID](https://orcid.org/0000-0001-8447-7145) <https://orcid.org/0000-0001-8447-7145>)

Kadir Furkan ÇAYBAŞI, Department of Mechatronics Engineering, Istanbul Technical University, Turkey, caybasik18@itu.edu.tr

([ID](https://orcid.org/0000-0001-8200-4010) <https://orcid.org/0000-0001-8200-4010>)

Bekir ÖZTÜRK, Department of Mechatronics Engineering, Istanbul Technical University, Turkey, ozturkbe18@itu.edu.tr

([ID](https://orcid.org/0000-0003-0577-3929) <https://orcid.org/0000-0003-0577-3929>)

*Volkan SEZER, Department of Control and Automation Engineering, Istanbul Technical University, Turkey, sezerv@itu.edu.tr

([ID](https://orcid.org/0000-0002-1825-0097) <https://orcid.org/0000-0002-1825-0097>)

Received: 21.12.2019, Accepted: 19.07.2020

Research Article

*Corresponding author

DOI: 10.22531/muglajsci.660496

Abstract

In this paper, design and implementation of an autonomous robot that collects tennis balls autonomously are explained. A small-scale tennis field is modeled as a ball-gathering area. In order to obtain the required image for localization and planning, a camera is placed 2 meters above the ground surface. Vehicle control is achieved after the image processing via Matlab™ software. Tennis balls are mechanically guided to the launch mechanism using guides and are collected in the chamber. Experimental results show that the design successfully fulfills its function.

Keywords: Image processing, travelling salesman problem, autonomous robot, tennis ball collector

TENİS TOPU TOPLAYAN OTONOM ROBOT TASARIMI

Özet

Bu çalışma kapsamında, tenis topunu otonom olarak toplayabilen bir robotun tasarımı ve gerçekleşmesi anlatılmaktadır. Top toplama alanı olarak, küçük ölçekli bir tenis sahası modellenmiştir. Planlama ve lokalizasyon için gerek duyulan görüntünün elde edilebilmesi için sahanın merkezine 2 metre yüksekliğinde bir kamera yerleştirilmiştir. Matlab™ yazılımı kullanılarak görüntü işleme gerçekleştirilmiş ve araç kontrolü sağlanmıştır. Geliştirilen robotta tenis topları, mekanik olarak kılavuzlar yardımıyla fırlatma mekanizmasına alınmakta ve haznedeki toplanmaktadır. Deneysel sonuçlar, yapılan tasarımın başarılı bir şekilde işlevini yerine getirdiğini göstermektedir.

Anahtar Kelimeler: Görüntü işleme, gezgin satıcı problemi, otonom robot, tenis top toplayıcı

Cite

Sarecalar, D., Çaybaşı, K. F., Öztürk, B., Sezer, V., (2020). "Development of an autonomous tennis ball retriever robot", *Mugla Journal of Science and Technology*, 6(0), 36-44.

1. Introduction

Tennis is one of the most popular and entertaining sports today. However, if you have no balls left during the game, you must leave the game and collect the balls. Especially if you work with a coach, the consumed time for the collection of the balls is a financial loss. Similarly, at the end of the game, you have to collect hundreds of balls from the field. This is really difficult for players who are tired during the game. Different tennis ball collection tools have been developed to solve this problem.

The commonly used Tube Tennis Ball Pickup consists of a pipe and a ball collector at the end of the pipe. It is used manually and shown in Chapter 4. Since it is a manual tool, the player must take a break to collect the ball. There are similar versions of manual tools in the market. However, they are not able to provide a clear

solution to this problem despite the shortening of the ball collection time.

In one of the tennis ball collector robot design, the ball collecting mechanisms rotate in the form of rollers [1]. However in this design, when the balls are taken into the storage, balls may squeeze in the system. The capacity of the collecting chambers of these robots is also low. Another ball collector robot solution [2] is provided based on a conveyor system. Although its ball collection capacity is high, it is heavy and not portable. In the robotic arm mechanisms for ball collection such as illustrated in [3], if there are lots of balls in the field, the robot must stop and set its position to take each of the balls. So it takes long time to collect all the balls from the field with such a solution. The complexity and cost are other disadvantages of robotic arm solutions. On the other hand, it is seen that a throwing mechanism collects the balls successfully in another design called "Tennibot"[4]. However, since the camera is on the

robot, the position of all the balls in the field can be hard to found.

The camera system which is used in the robots in previous works, is different than our proposed design. Instead of using on-board camera we use a static camera in our solution. So the camera's height is independent from robot body and can be adjusted independently. Since the camera is above the field, the robot is able to see the balls from bird's eye perspective. In this way, the robot is able to see more balls and this provides a more efficient optimization for robot route calculation. Additionally, since the robots can be seen from top, there is no need for self-localization. Using distinctive specific markers on top of the robots, ground station is able to understand the pose of each robot in the system.

Another contribution of our system is coming from proposed overall system architecture. Instead of on board processing, we use a ground station for data processing and decision making as shown in Figure 1. Since heavy processing requirement for image processing is transferred to a ground station, on-board computational hardware requirement reduces and a cheap microprocessor can be used on the robot. In addition, multi-agent applications for ball collection can be applied easier in this architecture. Since the robot does not have on-board camera and powerful computers, the robot itself is cheaper than the classical solution. However, the ground-station and communication need of robots bring extra cost to the overall system. As a result, the classical approach where the camera and all processing computers are on the robot, would be more effective if just one collector robot is enough. On the other hand our proposed solution will be more advantageous as the number of robots increases when a fast ball collection is required.

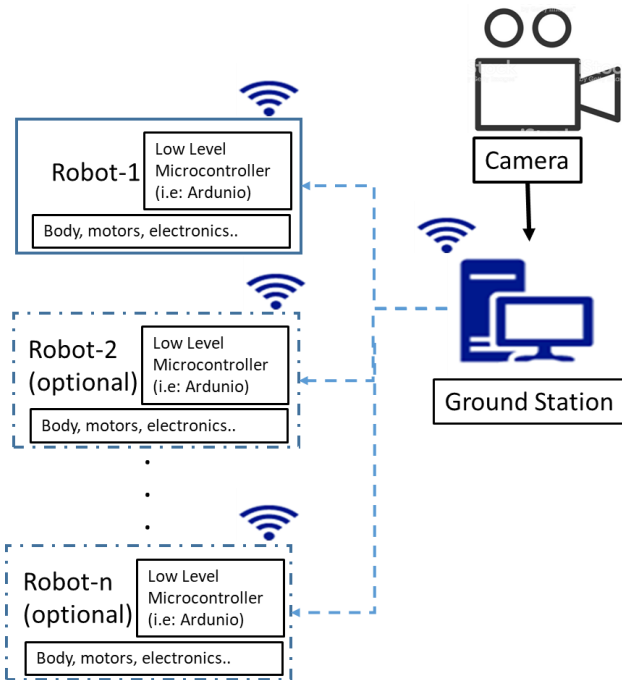


Figure 1. Proposed System Architecture

Rest of the paper is organized as follows; Chapter 2 explains the image processing and determination of target angle of the robot. In chapter 3, the implementation of the travelling salesman problem for the ball collection problem is provided. Chapter 4 describes the mechanical design of the system. Electronics, control and communication systems are explained in chapter 5. The paper ends with the experimental results and conclusions in Chapter 6.

2. Image Processing and Determination of Target Angle

The first idea of image processing was based on the cable transfer of images for the media industry in the 1920s. This was achieved with the Bartlane cable system, which enabled the transfer of images between London and New York in less than 3 hours. These images, which can show 5 shades of gray, reached 15 shades of gray in 1929 [5]. We can define image processing as a form of signal processing for which the input is an image, such as photographs or frames of video; the output of image processing can be either an image or a set of characteristics or parameters related to the image. There are a large number of applications of image processing in a diverse spectrum of human activities, from remotely sensed scene interpretation to biomedical image interpretation [6]. These applications can be done by using several techniques of image processing. Image preprocessing, image enhancement, image segmentation, feature extraction, and image classification are some of these techniques [7].

Image segmentation is a processing technique which is used to analyze the image. It can be defined as a processing technique to cluster an image into several disjoint parts by grouping the pixels to form a region of homogeneity based on the pixel characteristics [8]. The efficiency of segmentation process is its speed, good shape matching and better shape connectivity with its segmenting result. Table 1 compares the image segmentation techniques from different perspectives such as speed, noise immunity and accuracy [9]. As a result of the comparison given in Table 1, the threshold based segmentation method is selected since it is fast and has less complexity.

Table 1. Image segmentation technics [9]

Parameter	Threshold Based Segmentation	Edge Based Segmentation	Region Based Segmentation
Nature of the Output Image	Black-White	Black-White	Black-White
Spatial Information	Neglected	Neglected	Considered
Region-Continuity	Moderate	Moderate	High
Computation Complexity	Less	Moderate	High
Speed	Fast	Moderate	Slow
Noise Immunity	Less	Less	Less
Detection of Multiple Object	Poor	Poor	Moderate
Automaticity	Interactive (Semi Automatic)	Interactive	Interactive (Semi Automatic)
Accuracy	Moderate	Moderate	Good

In this paper, the position of the balls and robot are determined by using a color separation method on the image taken from the camera located on the top of the field. Figure 2 shows the block diagram of the general operation of the system.

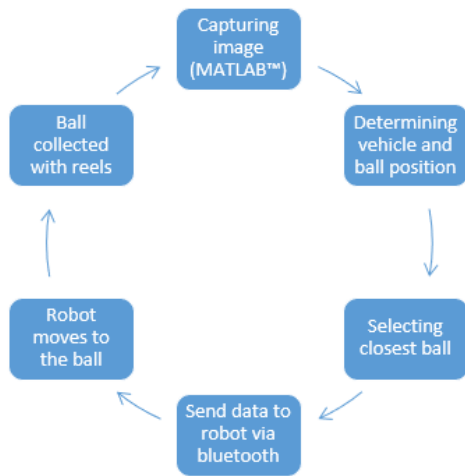


Figure 2. Block diagram of system operation.

To determine the position of the robot, two blue circles, one larger than the other, are placed on the robot. On the image, the blue color is masked and circles are detected in the image masked against the blue. As shown in Figure 3, the position of the robot is found with the help of these circles, the centers of the circles are combined and a line is obtained. Using this line, the orientation of the robot relative to the global axis is obtained successfully.



Figure 3. Determining robot position.

In order to understand the position of the balls; green color in the image is masked and circles in the image is found. As it can be understood from Figure 4, the centers of the balls are same as the centers of the circles.

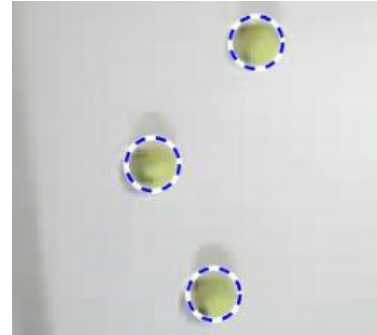


Figure 4. Determining the position of the balls.

The large blue circle in Figure 2, which is placed onto the rear axle of the robot is designated as the center of rotation of the robot. A local coordinate system is formed in the center of the large blue circle to enable the robot to orient towards the ball.

In order to calculate the rotating angle for the root to head on the balls, straight lines are drawn between the local coordinate system and the center of the balls. As it is shown in Figure 5 shows the required rotating angles of the robot for each of the balls. Similarly as shown in Figure 6, the red lines show the linear paths to the balls calculated on a real image and blue lines show the overall path. Calculation of this path is explained in the following chapter.

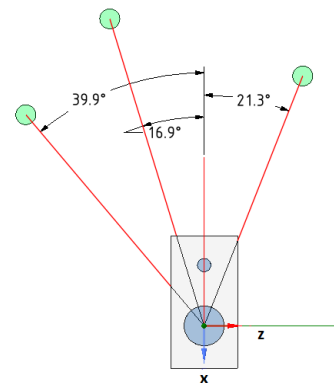


Figure 5. Determining the target angle.

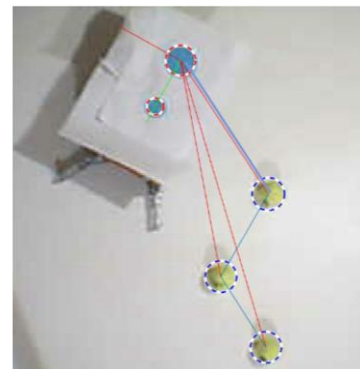


Figure 6. Distance to each of the balls and the target path.

3. Implementation of Travelling Salesman Problem into Road Planning

Different methods have been used to plan the motion of the ball collector robots for covering the whole field. Sweeping based method [10] or more advanced path planning approaches are available [11]. On the other hand, it is also very important to plan in which order the balls will be collected. The traveling salesman problem (TSP) is very suitable to define ball collection problem. TSP was formulated as a mathematical problem in the 1930s. It is one of the deepest issues in optimization. The problem can be identified by the following steps [12]

- There is a salesman
- This seller wants to sell his/her goods in “n” cities
- The seller wants to tour these cities as soon as possible and to visit each city only once

TSP has been used in various applications, such as drilling circuit boards, overhauling gas turbine engines, computer wiring and vehicle routing as in this paper [13] [14]. Although it is easy to define the traveling salesman problem, it is very difficult to obtain the optimal solution especially when the number of “n” is high, and it is among the NP-hard problems in the literature. As these numbers grow larger, it becomes harder for a computer to solve such problems in a reasonable amount of time. Table 2 illustrates the growth of maximum Hamiltonian Cycles for various number of vertices in TSP. Solution of the TSP needs specific algorithms in order to make computations in acceptable time [15].

Table 2. Effect number of cities on Number of Cycles[15]

n	(n-1)!/2
3	1
4	3
5	12
6	60
7	360
8	2,520
9	20,160
10	181,440
20	6.08226E+16
30	4.42088E+30
40	1.01989E+46

By integrating the TSP problem into this study; the robot is intended to collect the balls as soon as possible and extend the battery life. In this study, “Closest City Algorithm” and “Genetic Algorithm” methods are examined for the solution of the problem.

“The Closest City Algorithm” is the application of the nearest neighbor algorithm for the solution of TSP [16] and it is based on the following very easy principle: Every time, the salesman prefers the closest city which

has not been visited yet. Although the closest city approach does not guarantee the optimal solution, it is frequently used for TSP problem because of its simplicity. Pseudo code of the closest city algorithm is illustrated below:

1. select a city as current city.
2. find out the shortest edge connecting the current city and an unvisited city.
3. set the new city as current city.
4. mark the previous current city as visited.
5. if all the cities are visited, then terminate.
6. Go to step 2.

The length of the route drawn for 10 balls by this method is 3993 units as shown in Figure 7. The solution time is approximately 1.2 seconds. Computer is used that has CPU and RAM properties CORE İ7 4720HQ 2.6GHZ and 16GB DDR3 1600 MHz. The algorithms are tested on MATLAB software.

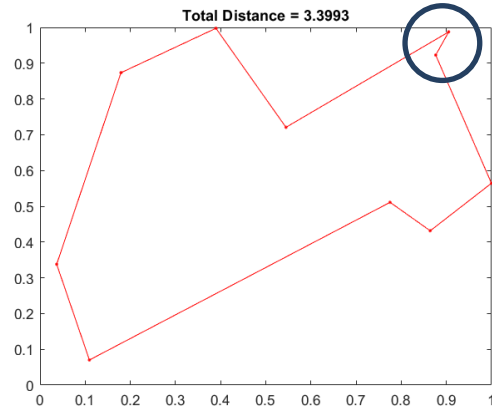


Figure 7. The Closest City Algorithm Solution.

On the other hand, genetic algorithm is an intuitive search technique based on parameter coding which tries to find solutions using random search techniques. The key characteristic of the genetic algorithm is its searching process. The algorithm creates a “population” of possible solutions to the problem and lets them “evolve” over multiple generations to find better and better solutions [17]. This method cannot guarantee the optimal solution, but it can achieve a reasonable solution in a short time. Genetic algorithms are particularly suitable for solving difficult optimization problems where conventional optimization methods are less effective [18].

With this method, the length of the route drawn for 10 balls is 3937 units. The solution time is approximately 2.6 seconds.

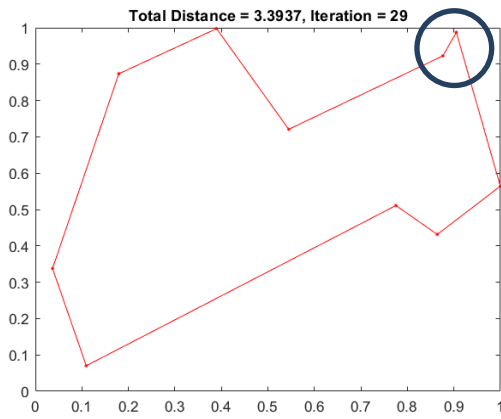


Figure 8. Genetic Algorithm Solution.

According to the results shown Figure 7 and Figure 8, there is very low difference between paths which are circled in each of the figures. This small difference results with 1.5% reduction of the total path but the increment of required time becomes more than two times. When the solution time and road lengths are evaluated, it is thought that the solution will be more advantageous with “The Closest City Algorithm”. Especially, when the number of balls in the field increases, it is seen that “Genetic Algorithm” method will create disadvantage in terms of processing time.

4. Mechanical Design

Collecting tennis balls is a tiring and time-consuming task for players, especially during training. Different tennis ball collection methods have been developed to make this boring job easier.

Figure 9 illustrates two different tools for manual collection of tennis balls. However, this type of solution requires the user to stop the training session to collect the balls. Although these tools are good in terms of cost and portability, speed and consumed time are the main disadvantages.



Figure 9. Tennis ball collection with manual tools [3].

In this paper, the following different designs have been analyzed in order to meet the needs in the best way. The system shown in Figure 10 uses a rotary shaft mechanism which does not need an extra actuator for ball collection [1]. However, in this method, when the balls are taken into the storage, balls may squeeze in the system. A similar situation can be observed in the driven rotary shaft mechanism shown in Figure 11 [19]. In addition, the number of balls it can collect is limited

because the balls are stored side by side in the storage. The robotic arm design shown in Figure 12 is a complicated mechanism comparing to the others hence it is not an efficient way to collect the balls. There are lots of balls in the field and the robot must stop and take a position to take each of the balls [3]. Conveyor mechanism is used in the system shown in Figure 13. Tennis balls are thrown into a basket with the help of a conveyor [2]. 80 balls can be collected per minute. Between 120 and 140 balls can be stored in the basket. Although this method seems to be fast, solar energy systems and hydraulic mechanisms make it expensive. Since the dimension of the design is 105x107cm it is not portable enough. In addition, the mechanisms used in the system increases the weight of the robot too much. In the throwing mechanism shown in Figure 14, more balls can be collected because the balls are stacked on top of each other [4]. Since this system is simple and less problematic, the ball collector mechanism is designed using this throwing mechanism.



Figure 10. Rotary shaft mechanism without drive [1].



Figure 11. Rotary shaft mechanism with drive [19].



Figure 12. Robot arm mechanism [3].

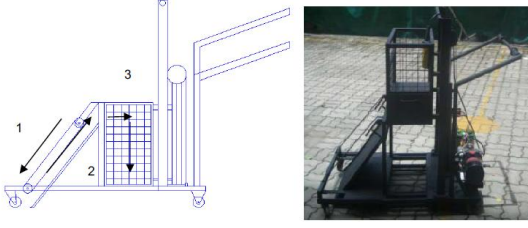


Figure 13. Conveyor mechanism [2].



Figure 14. Throwing mechanism [4].

These studies in the literature were examined in detail and Table 3 is created. As a result, ball throwing mechanism is selected for our robot. The throwing mechanism designed for our robot is explained in detail, in next section.

Table 3. Comparison of collector mechanism

	Complexity	Weight	Efficiency	Cost
Manual tools	Very Low	Medium	Medium	Low
Rotary shaft mechanism without drive	Low	Low	Low	Medium
Rotary shaft mechanism with drive	Medium	Medium	Medium	High
Robot arm mechanism	Very High	High	Very Low	High
Conveyor mechanism	High	Very High	Low	Very High
Throwing mechanism	Medium	Medium	Very High	High
	Low is better	Low is better	High is better	Low is better

4.1. Throwing Mechanism

In a throwing mechanism, tennis balls are directed to the system with the help of the guides. Balls thrown from the throwing mechanism are collected in the storage. Firstly, a risk assessment is performed by creating a design substructure to meet the needs of the system. As a result, a test prototype was constructed to determine the roll slope and height of the reels.

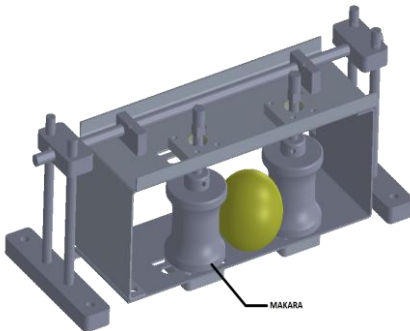


Figure 15. Prototype test design.

In the prototype shown in Figure 15, optimum design criteria are determined by experimenting with different reel designs and slope angles.

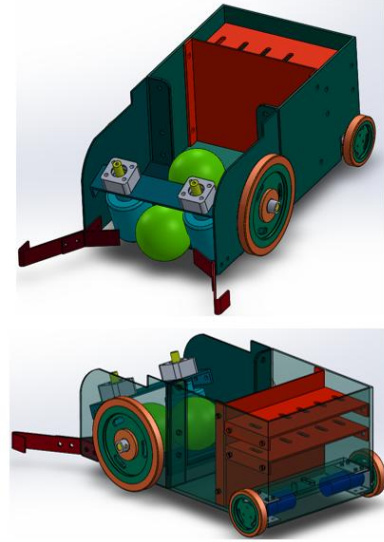


Figure 16. Robot front and rear view.

After the final design which is shown in Figure 16, mechanism is produced using 3D printer and it is observed that the collection of the balls are performed successfully. With the help of guides, the tennis balls can be centered moving towards the reels. The real production of the designed system is shown in Figure 17.

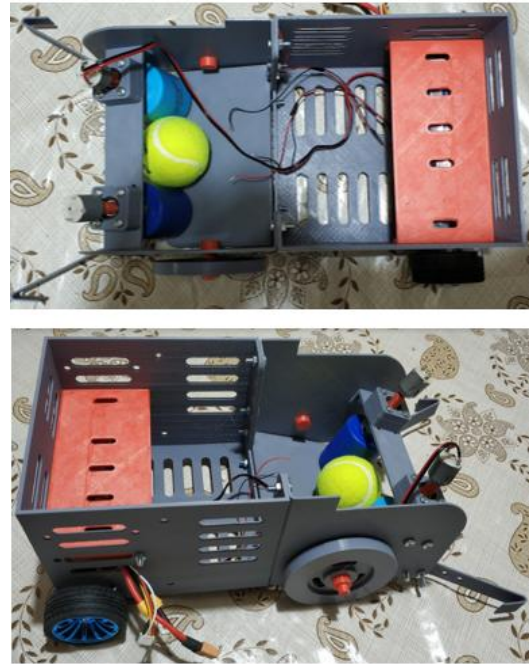


Figure 17. 3D printed throwing mechanism and body.

5. Electronics, Control and Communication System

In this study system is designed to collect the balls by executing the commands coming from Matlab™ after image processing process. The electronic materials required to accomplish this task are given below and the purpose of each component are described accordingly.

- Arduino NANO
- Bluetooth Module
- L298n Motor Driver Board
- 5V Relay
- Voltage Booster Regulator
- Voltage Reducer Regulator
- 2 pieces of 12 Volt 100 RPM DC Motor
- 2 pieces of 24 Volt 200 RPM DC Motor
- 12V 5000mAh Li-Po Battery
- A4Tech Webcam

The Arduino NANO communicates with the computer through its bluetooth module. It controls the traction motors and reels as ordered by the commands from the computer. The L298n motor driver board enables the motor to move according to the PWM signals sent from the control board. The relay enables the roller motors to operate in a controlled way. The DC-DC step-up converter increases the voltage in the battery and provides it to the motors of the collecting mechanism. The DC-DC step-down converter reduces the voltage in the battery to the voltage required for the Arduino NANO to operate. The camera can take up to 8 pictures per second and is compatible with Matlab™ software. The battery provides the energy required for the operation of the whole system and can operate the system for 6 hours without any interrupt.

The operating principle of the system is provided in Figure 18 as a block diagram. Matlab™ software runs on the computer and processes the image from the camera. After determining the position of the robot and the position of the balls, the commands needed for the robot to catch the ball are transmitted to the controller card by communicating via bluetooth module.

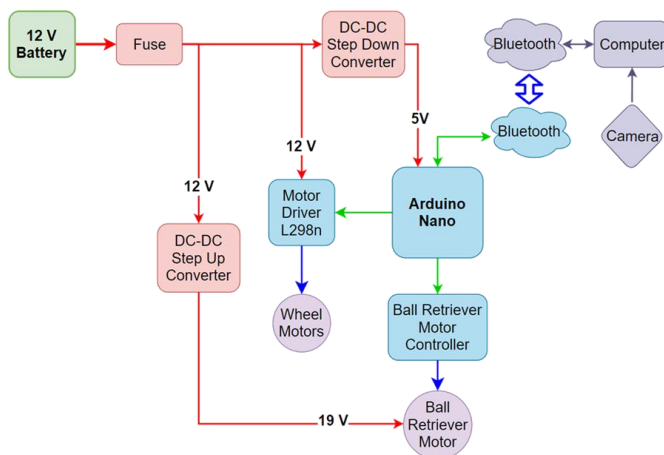
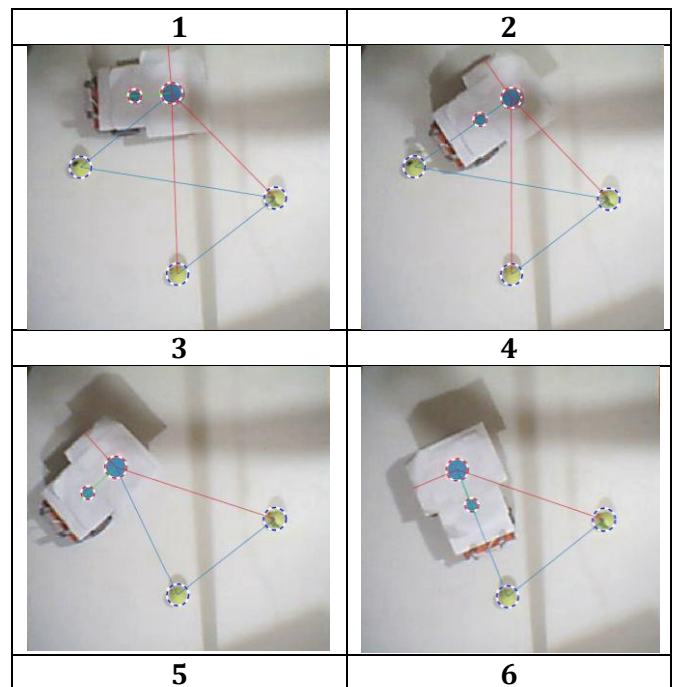


Figure 18. Block diagram of electronic components.

6. Experimental Results and Conclusions

This paper presents the design, production and autonomous work of an autonomous robot that collects balls. Thanks to the designed throwing mechanism, the balls are collected efficiently without jamming. With the communication system established between the ground station and the vehicle, the need for the use of equipments such as sensors for localization or perception and embedded computers on the robot are eliminated. By using image processing, the robot moves towards the position of the balls. The route is determined using the Nearest City Algorithm, which is one of the methods to solve the traveling salesman problem. As image processing is performed via Matlab™ on a remote computer, the necessary information is transferred to the robot wireless. When the robot reaches the required position, the pulley system is used to convey the balls smoothly to the storage. Figure 19 and Figure 20 show two different scenarios where the robot selects and collects the nearest ball respectively.



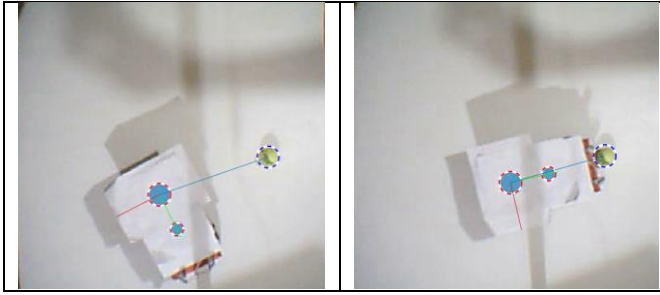


Figure 19. Real-time ball collection Scenario-1.

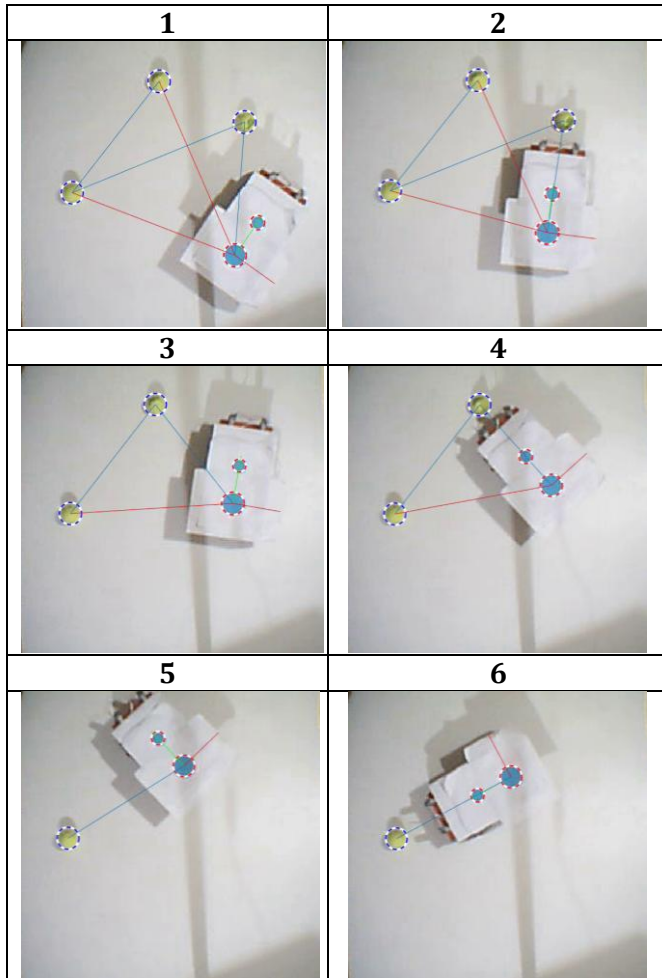


Figure 20. Real-time ball collection Scenario-2.

In this study, we provide a different solution where we have a ground station, separated camera and a collector robot as illustrated in Figure 1. In this way, the ground station is able to see more balls and this provides a more efficient optimization for robot route calculation. Moreover, in our approach, the collector robot(s) do not require on-board camera or high level processing computers inside. As a result, classical approach would be more efficient for just one robot while our proposed solution is more efficient as the number of robots increases for fast collection. A deeper analysis for the comparison of each approach in terms of ball collection speed, cost of the system and efficiency can be done as a future work.

Improvements can be made to the robot by adding path tracking algorithms and a higher resolution camera to scan a larger area and applying more functional image processing methods. In addition, the chamber of the existing prototype can be enlarged so that it can collect more balls at the same time. In order to avoid the possible obstacles in the field, several obstacle avoidance methods such as [20,21] can be applied on the robot in the future. As a future study, the system can be tested using multiple robots as a multi-agent application.

Detailed working video of the robot can be found at the link below.

<https://www.youtube.com/watch?v=G0G8x4Hck9s>

7. Acknowledgment

We would like to express our gratitude to the ITU Mechatronics Training and Research Center for supporting this study within the scope of MKM512-Mechatronics System Design course and to Alev Keskin.

8. References

1. Çabuk, Vehbi Umur, et al. "Design and Control of a Tennis Ball Collector Robot." 2018 6th International Conference on Control Engineering & Information Technology (CEIT). IEEE, 2018.
2. Am-Eam, Nanthawan. "The implementation of hydraulics system and solar energy in the Tennis Balls Collecting Machine." ICCAS 2010. IEEE, 2010.
3. Chawla, Neha, et al. "Robotic Tennis Ball Collector." Florida Conference on Recent Advances in Robotics. 2012.
4. <https://www.tennibot.com/>
5. Shinde, Bhausahab Shivajirao, and A. R. Dani. "The Origins of Digital Image Processing & Application areas in Digital Image Processing Medical Images." IOSR Journal of Engineering 1.1: 066-071, 2007
6. Acharya, Tinku, and Ajoy K. Ray. Image processing: principles and applications. John Wiley & Sons, 2005.
7. Chitradevi, B., and P. Srimathi. "An overview on image processing techniques." International Journal of Innovative Research in Computer 2.11, 6466-6472, 2014
8. Khan, A. M., and S. Ravi. "Image segmentation methods: A comparative study." (2013).
9. Pal, Nikhil R., and Sankar K. Pal. "A review on image segmentation techniques." Pattern recognition 26.9 (1993): 1277-1294.
10. Elamvazuthi, I., et al. "Development of an autonomous tennis ball retriever robot as an educational tool." Procedia Computer Science 76 (2015): 21-26.
11. Ali, M. F., et al. "Design and Development of Path Planning Techniques for a Tennis Ball Retriever Robot." Journal of Telecommunication, Electronic

- and Computer Engineering (JTEC) 10.1-15 (2018): 59-65.
12. Sural, H. "Gezgin Satıcı Problemi." *Matematik Dünyası dergisi*, 37-40, 2003
 13. Matai, Rajesh, Surya Prakash Singh, and Murari Lal Mittal. "Traveling salesman problem: an overview of applications, formulations, and solution approaches." *Traveling salesman problem, theory and applications 1* (2010).
 14. Gu, Shenshen, et al. "A deep learning tennis ball collection robot and the implementation on nvidia jetson tx1 board." *2018 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE, 2018.
 15. Corinne Brucato "The travelling salesman problem" Thesis Master of Sciences University of Pittsburgh 2013.
 16. Kizilates, Gözde, and Fidan Nuriyeva. "On the nearest neighbor algorithms for the traveling salesman problem." *Advances in Computational Science, Engineering and Information Technology*. Springer, Heidelberg, 111-118, 2013.
 17. Thede, Scott M. "An introduction to genetic algorithms." *Journal of Computing Sciences in Colleges* 20.1 (2004): 115-123.
 18. Pulat, M., DEVECİ KOCAKOÇ, İ. (2016), Gezgin Satıcı Probleminin Genetik Algoritmalar Kullanarak Çözümünde Çaprazlama Operatörlerinin Örnek Olaylar Bazlı İncelenmesi, Yöneylem Araştırması Endüstri Mühendisliği 36. Ulusal Kongresi, BORNova, 2016.
 19. Chen, H.-K., & Dai, J.-M. (2016). An Intelligent Tennis Ball Collecting Vehicle Using Smart Phone Touch-Based Interface. *2016 International Symposium on Computer, Consumer and Control (IS3C)*. doi:10.1109/is3c.2016.100
 20. Sezer, V. and Gokasan, M., "A novel obstacle avoidance algorithm: "Follow the Gap Method"". *Robotics and Autonomous Systems*, 60(9), 1123-1134, 2012.
 21. Fox, D., Burgard, W. and Thrun, S., "The dynamic window approach to collision avoidance" *IEEE Robotics & Automation Magazine*, 4(1), 23-33, 1997.