

Intelligent Robot Tennis Training based on YOLO

Hafid Al'azzah

Department Of Electrical Engineering, Faculty Of Vocational Studies, Universitas Negeri Surabaya, Indonesia
e-mail address: hafid.20001@mhs.unesa.ac.id (corresponding author)

Received: 16 September 2025 | Revised: 18 October 2025 | Accepted: 30 October 2025

This is an open access article under the [CC BY-SA](#) license.



ABSTRACT

Technological developments worldwide have had a significant impact on various aspects of life, including sports. One of the most popular sports is tennis. However, the use of technology in sports training is still limited, especially in the implementation of automated systems that can improve the efficiency and effectiveness of athlete training. Therefore, in this study, a smart tennis ball launcher based on You Only Look Once (YOLO) as an image processing technology was designed. This tool aims to provide new innovations in athlete training, while also assisting coaches in developing more effective training programs. This system uses several hardware components such as Arduino UNO, DC Motor, Servo Motor MG995, Driver BTS7960, and a camera as the main sensor. Meanwhile, the software used includes Arduino IDE to control the hardware, and Visual Studio Code with the Python programming language to implement the YOLO algorithm in detecting player positions. The results of this study show that the device is able to launch tennis balls to four predetermined areas of the court, according to the player's position detected by the camera. Thus, this tool can be an automated solution in more accurate and efficient tennis training.

Keywords: *YOLO; Image Processing; Python Programming; Object Detection; Arduino UNO*

1. INTRODUCTION

Tennis is a sport played on a rectangular court, with two players (singles) or two pairs (doubles) facing each other. Players use rackets to hit the ball over the net and bounce it back to the opponent's side to score points. Victory in a tennis match is determined by a player's ability to control the ball and place it in a difficult-to-return area of the opponent's court. Therefore, intensive and innovative training is essential to improve athletes' technical and tactical skills. With the advancement of technology, the world of sports has also begun to utilize technological innovations to improve training effectiveness [1][2][3][4][5]. However, in tennis training, the application of automation technology is still limited [6][7][8][9][10]. Generally, existing ball-launching devices have limitations in terms of ball direction and speed variation, and do not utilize real-time player position detection systems. This leads to a lack of personalization in training programs, thus not fully supporting improved athlete performance.

Image processing is a technique for processing digital images using a computer so that they can be analyzed, modified, or utilized for specific applications such as object recognition, face detection, or image enhancement [11][12][13][14][15]. The main objectives of image processing are to improve image quality (e.g., remove noise or clarify details), extract important information from images (e.g., recognize objects, patterns, or shapes), and transform images for technical purposes, such as rotation, scaling, segmentation, or color conversion [16][17][18][19][20].

Therefore, this study designed a smart tennis ball-launching device based on the You Only Look Once (YOLO) system. This tool is designed to provide more dynamic and specific training variations based on the player's position and movement, detected in real-time via the camera. By utilizing the YOLOv3 algorithm, the system is able to accurately detect the player's position and automatically control the direction and speed of the ball throw. This system consists of several hardware components, such as an Arduino UNO, a DC motor, an MG995 servo motor, a BTS7960 driver, and a camera as the primary sensor. The software used includes the Arduino IDE for hardware control and Visual Studio Code (Python) for implementing the YOLOv3 object detection algorithm. Research Contribution:

- Technological Innovation in Tennis Training: Development of a ball thrower that can adapt to the player's position in real-time.
- Improved Training Effectiveness: The tool is capable of throwing the ball to four predetermined areas of the court, based on the player's position detected by the camera.

- Training System Automation: Provides more consistent and varied training patterns, helping athletes improve their fundamental technical skills and reactions to balls coming from various directions.
- AI-Based Approach: Utilizes the YOLOv3 image processing algorithm to support an accurate and responsive object detection system.

This automation system is expected to provide an innovative solution for more effective, efficient, and personalized tennis training, as well as assist coaches in developing more targeted and data-driven training programs.

2. METHOD

2.1. YOLO

YOLO (You Only Look Once) is a deep learning-based object detection algorithm known for its speed and accuracy [21][22][23][24]. Unlike previous object detection approaches such as R-CNN and Fast R-CNN, which operate in stages (region detection and then classification), YOLO performs object detection in a single process (end-to-end). This algorithm views the image as a whole and immediately predicts the bounding box and class label for each object detected in the image [25][26][27][28][29][30]. YOLO has the following advantages:

- Fast: Suitable for real-time applications, such as surveillance, drones, and autonomous cars.
- Accurate: Has high accuracy.
- Efficient: Requires fewer computational resources than two-stage detection methods.

2.2. Python

Python is a high-level, open-source programming language designed to be easy for humans to read and write [31][32][33][34][35]. Developed by Guido van Rossum and first released in 1991, Python is very popular due to its simple syntax, making it suitable for both beginners and professionals [36][37][38][39][40]. Python is designed with the philosophy of ease of use and code readability, making it ideal for both beginners and professionals [41][42][43][44][45]. Python's characteristics and advantages include:

- Simple, English-like syntax.
- Extensive community support and rich documentation.
- Interpretive language: no need to compile; just run.
- Portable: can run on Windows, macOS, Linux, etc.
- Open source: free and can be developed by anyone.
- Large library support: such as NumPy, Pandas, TensorFlow, Django.

2.3. Microcontroller

A microcontroller is a small computer chip (IC - Integrated Circuit) that contains a processor (CPU), memory (RAM and ROM/Flash), and input/output (I/O) in a single package [46][47][48][49][50]. Microcontrollers are used to automatically control electronic devices, such as washing machines, TV remotes, alarm systems, robots, and IoT devices. Example of popular microcontroller can be seen in Table 1. Main Microcontroller Components

- CPU (Central Processing Unit): The main brain that executes program instructions.
- RAM: For temporary data storage while the program is running.
- ROM/Flash: Stores permanent user-written programs.
- I/O Ports: For interacting with the outside world (sensors, motors, LEDs, etc.).
- Timers/Counters: For calculating time, delays, or counting pulses.
- ADC (Analog to Digital Converter): Converts analog signals (e.g., from a temperature sensor) to digital.
- Serial Communication: Such as UART, SPI, and I2C for communicating with other modules.

TABLE 1. EXAMPLES OF POPULAR MICROCONTROLLERS

Mikrokontroler	Information
ATmega328P	Arduino Uno
ESP32 / ESP8266	Supports Wi-Fi & Bluetooth, ideal for IoT
STM32	High performance, used in industry and real-time systems
PIC	Widely used in industry and automation
RP2040	Chip from the Raspberry Pi

2.4. DC Motor

A DC motor (direct current motor) is an electric motor that converts direct current (DC) electrical energy into mechanical energy [51][52][53][54][55]. DC motors are used in various devices such as fans, robots, toys, electric vehicles, and automatic control systems. Parts of a DC Motor:

- Armature : a rotating coil through which current flows.
- Stator : the part that produces a permanent magnetic field.
- Commutator : a two-ringed ring that changes the direction of the current in the armature.
- Brushes : carbon or metal parts that connect the power source to the commutator.
- Shaft : the point where mechanical power (rotation) is output.

Types of DC Motors

- Brushed DC Motor: Common and inexpensive. Uses a commutator and brushes. More frequent maintenance due to brush wear.
- Brushless DC Motor (BLDC): Brushless; more efficient and durable. Used in drones, computer fans, and electric vehicles.
- Control is more complex (using an electronic controller). Servo Motor (DC-based), Equipped with a position control system and Suitable for robotics and precision control.
- Geared DC Motor: Equipped with a gearbox to reduce speed and increase torque.

2.5. Switched-Mode Power Supply (SMPS)

A Switched-Mode Power Supply (SMPS) is an electronic power supply that converts electricity from one form to another with high efficiency using switching components (such as transistors) that operate at high speeds (on/off) [56][57][58][59][60]. SMPSs are used in various electronic devices such as computers, TVs, cell phone chargers, printers, and industrial systems. The function of an SMPS is to convert and regulate input voltage (AC or DC) into a stable output voltage (DC), with high efficiency and a smaller physical size than a linear power supply. The advantages of an SMPS include: High efficiency (>80%) due to minimal energy wasted as heat, Smaller and lighter size than a linear transformer, Flexible output voltage (can be step-up, step-down, or isolated), Easier cooling due to less heat. Types of SMPS include Buck Converters (step-down voltage), Boost Converters (step-up voltage), Buck-Boost Converters (increase or decrease voltage), Flyback Converters (common in adapters and chargers), Forward Converters, Push-Pull Converters, Half-Bridge Converters, and Full-Bridge Converters (used in high-power SMPSs). The Figure 1 is SMPS flow diagram

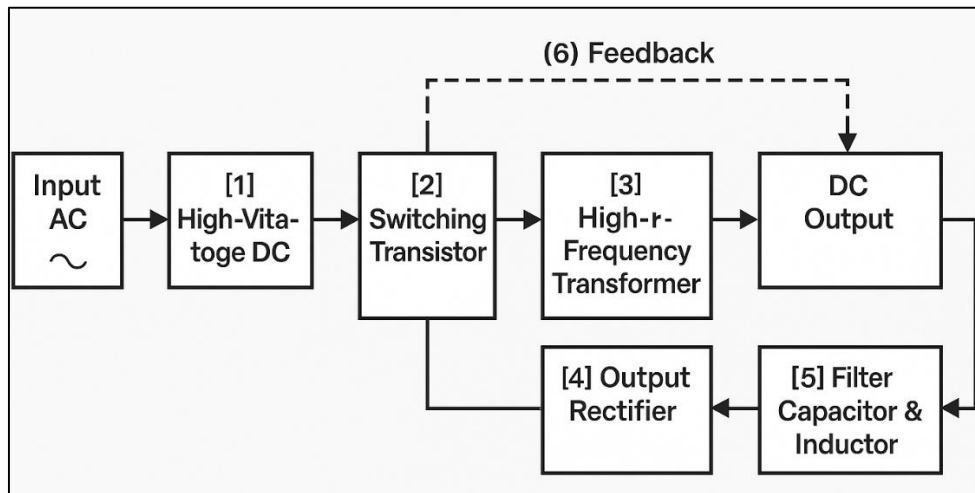


Figure. 1. SMPS flow diagram

3. PROPOSED METHOD

This study uses a system design method to develop a smart tennis training device based on YOLOv3 and a microcontroller. The design consists of hardware (Mechanical & Electrical) and software design. The hardware consists of an Arduino Uno, a JGY-370 DC motor, an RS-775 DC motor, a BTS7960 motor driver, a 12V 20A SMPS, and a smartphone camera as a visual sensor. The software was developed using Python with the YOLOv3 library for object detection and the Arduino IDE for

microcontroller programming. The goal of this research is an automatic control system for a tennis training device capable of responding to player movements based on visual detection from the camera. The study was conducted on a prototype device tested in a simulated tennis court environment. The software used is: (1) VS Code, (2) Arduino IDE, (3) Python. Meanwhile, the hardware used is: (1) Arduino Uno, (2) RS-775 DC Motor, (3) JGY-370 DC Motor, (4) BTS7960 Motor Driver, (5) 12 Volt 20 Ampere SMPS, (6) Camera, (7) Laptop. The Detail of device can be seen in [Figure 2](#), [Figure 3](#) and [Figure 4](#).

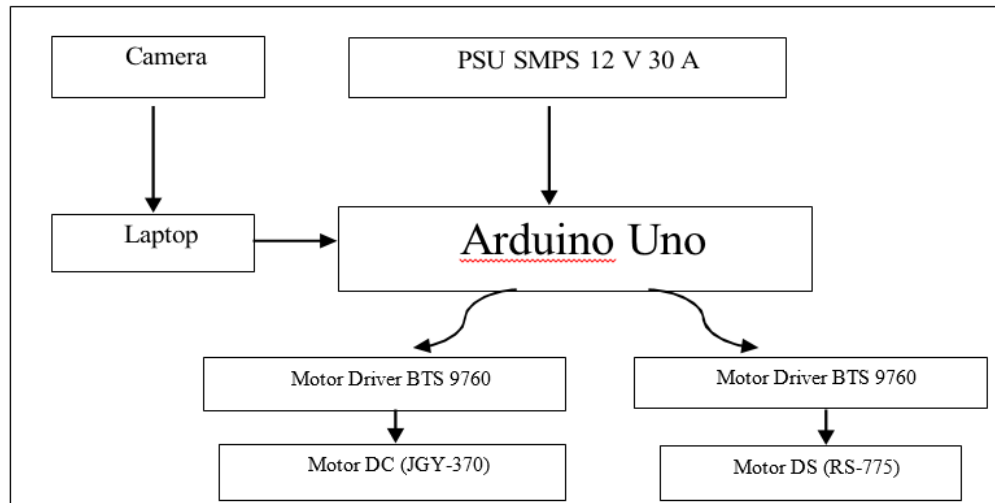


Figure 2. Overall System Block Diagram

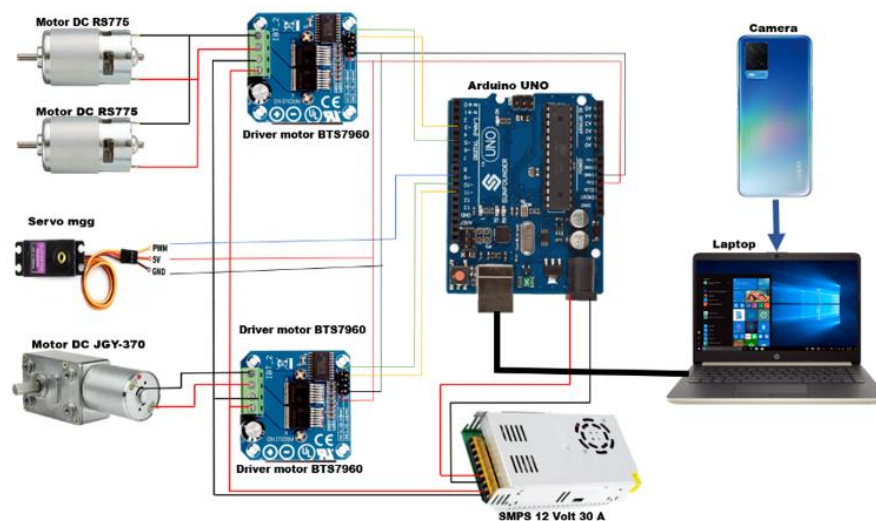


Figure 3. Hardware Design

Data was analyzed quantitatively based on mechanical test results, YOLOv3 detection accuracy, and control system response. Measured parameters included motor speed (RPM), player detection accuracy, and the system's effectiveness in adjusting the ball's throw direction based on the player's position on the court. This research aimed to test a microcontroller- and YOLOv3-based tennis ball throwing machine automation system through a series of tests. Mechanical testing was conducted to ensure the machine's frame assembly was in the correct position for proper function. Next, DC motor control testing was conducted to measure the time required for the motor to reach the ball's drop point, according to the designed program. Testing the YOLOv3 detection system was also a crucial part of this research, where the system was tested to detect objects on the tennis court and send signals to the microcontroller to trigger the machine's operation. The testing process consisted of several stages, including object detection using the YOLOv3 bounding box, motor RPM measurement, response time analysis for processing each frame, and control simulations that utilized YOLOv3 detection results for automated decision-making. Through these various testing stages, the system is expected to work optimally in supporting automatic tennis training and increasing the effectiveness of training for players.

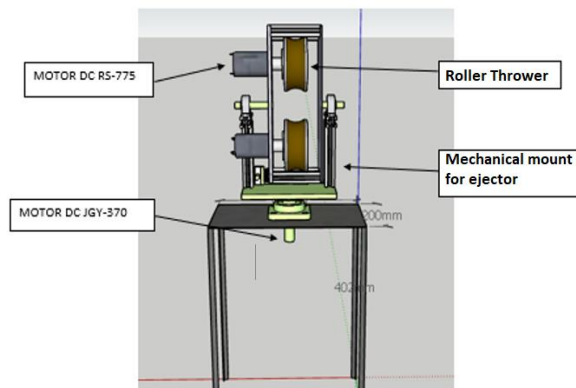


Figure 4. Hardware Design

4. RESULT AND DISCUSSION

Research on a tool or machine is conducted to determine its performance. The results can be analyzed into key points and conclusions can be drawn. The research and discussion chapter explains the testing process for the tool, including YOLO response testing and device testing. The goal of testing is to ensure the proper functioning of the automated tennis ball throwing machine. Therefore, accurate data is needed to compare the test results with manual calculations.

4.1 YOLO detection testing

This test aims to evaluate the accuracy and detection time of YOLO. Test results in four areas of the field showed good accuracy and an average detection time of 5-8 seconds. YOLOv3 successfully detected players or objects marked with a green box. When an object is detected, marked with a green box, the data is sent to the microcontroller to execute the ball throw. Objects tend to be easier to detect due to their larger size and clearer color contrast against the background. Below are images and the time required for positions 1-4. It can be seen in Figure 5.



Figure 5. YOLO response testing a) Position 1 b) Position 2 c) Position 3 d) Position 4

4.2 Device Testing

Hardware is any type of component that is physically visible or tangible on a device or machine. Hardware testing aims to determine whether the components in the device are functioning as expected. Hardware testing consists of mechanical testing and electrical testing.

4.3.1 Mechanical Testing

In the mechanical testing section, the operation of the frame assembly is examined in detail to ensure that the mechanical assembly is in the correct position and functioning properly. If the frame assembly is misaligned, the device will not function properly. The frame or body of this tennis ball throwing machine consists of iron plate, hollow iron, PVC, wood, and aluminum foil. For mechanical testing, it can be seen in [Figure 6](#).

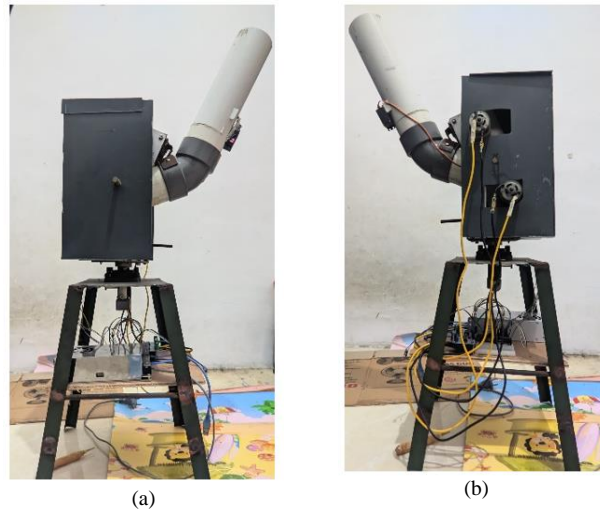


Figure 6. The throwing mechanism can move to a) the right b)left

4.3.2 Electrical Testing

This electrical test involves automatic control testing, with data collected in the form of distance and speed (rpm) required by the roller. Current and voltage testing, which measures the voltage and current values in the power circuit, and throw distance based on roller speed (rpm), are also performed. The ball throwing position can be seen in [Figure 7](#). Details of the speed data in the test can be seen in [Table 2](#).

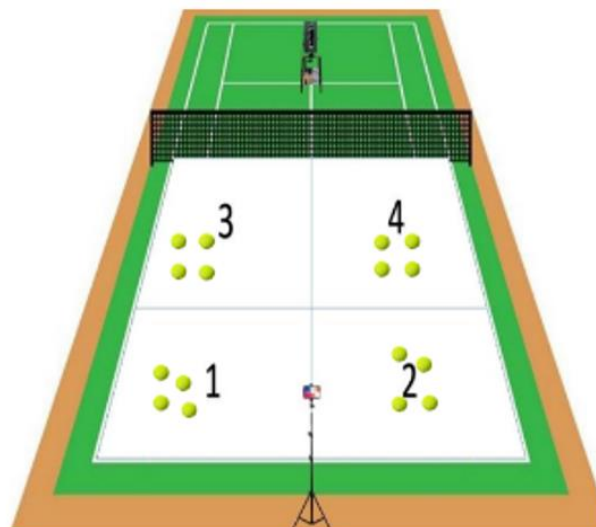


Figure 7. Ball Drop Mapping

TABLE 2. ROLLER SPEED TESTING





Testing Position	Ejection Speed (RPM)	Image
1	3616 3614 3612 3621 3622 3615 3610 3621 3612 3625	
2	3616 3614 3612 3621 3622 3615 3610 3621 3612 3625	
3	3439 3440 3435 3420 3425 3430 3437 3421 3429 3428	
4	3439 3440 3435 3420 3425 3430 3437 3421 3429 3428	

Table 3 shows that the balls thrown and the speed of the thrower have different values but tend to have the same pattern. Based on the distance the ball falls on the automatic control, the calculation of the distance the ball is thrown at the first point of fall can be found using the Pythagorean formula. However, the results of these calculations will be compared with the results of the actual distance measurements that occur during the testing of the tool. In the automatic system hardware testing, a single game is played with 4 balls, with the aim of reaching 4 points on the field. Table 5 shows that the number of balls thrown, the duration of the game, and the speed of the thrower have different values. The higher the level of the type of game, the higher the speed of the thrower. Based on the distance the ball falls for the automatic game, the calculation of how far the ball is thrown at the first point of fall can be found using the Pythagorean formula. The data obtained are as follows. As an explanation, the meaning of "measured" is the actual value of the distance measured on the field, while the "calculated" value is the value obtained from the Pythagorean calculation. It can be seen in Figure 8.

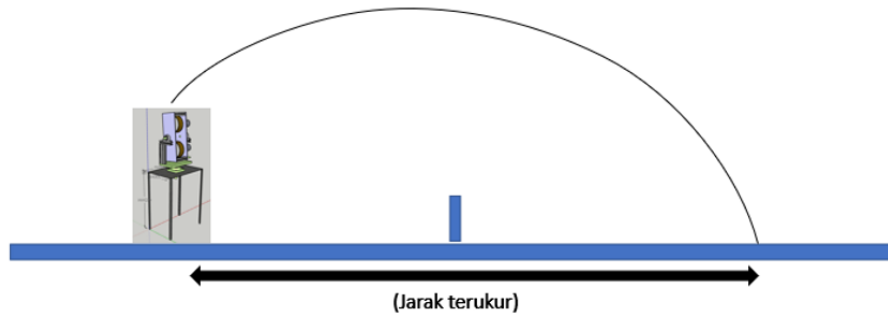


Figure. 8. illustration of tools with Pythagoras

TABLE 3. MAPPING THE DISTANCE THE BALL FALLS ON THE GRID

Field position	plot	Tilt angle	Ejection speed (RPM)	Measurable (m)	Counted (m)
1 and 2		24°	3616	12.9	16.73
			3614	12.8	16.73
			3612	12.8	16.73
			3621	12.9	16.73
			3622	12.9	16.73
			3615	12.8	16.73
			3610	12.8	16.73
			3621	12.9	16.73
			3612	12.8	16.73
3 and 4		24°	3439	9.8	16.35
			3440	9.7	16.35
			3435	9.7	16.35
			3420	9.6	16.35
			3425	9.7	16.35
			3430	9.8	16.35
			3437	9.8	16.35
			3421	9.6	16.35
			3429	9.7	16.35
			3428	9.7	16.35

The difference in roller speed values at several points on the field is due to the PWM (Pulse Width Modulation) value being increased for each ball drop. The 8-bit PWM resolution on the Atmega2560 Arduino Uno chip ranges from 0-255. In game positions “1” and “2” the PWM value used is 250, while in game positions “3” and “4” the PWM value used is 200. To calculate the duty cycle value and output voltage (Vout) of the PWM pin used can be written using the following formula.

Positions 1 and 2

$$\text{Duty cycle} = \frac{\text{PWM}}{255} \times 100\% = \frac{250}{255} \times 100\% = 98\% \quad (1)$$

$$\text{Vout} = \text{Duty cycle} \times \text{Vin} = 98\% \times 5 \text{ V} = 4.9 \text{ Volt} \quad (2)$$

Positions 3 and 4

$$\text{Duty cycle} = \frac{\text{PWM}}{255} \times 100\% = \frac{200}{255} \times 100\% = 78\% \quad (3)$$

$$\text{Vout} = \text{Duty cycle} \times \text{Vin} = 78\% \times 5 \text{ V} = 3.92 \text{ Volt} \quad (4)$$

Where Duty Cycle is Duty cycle percentage (0-255). Vout is PWM signal output voltage (volts). The SMPS power supply produces a voltage of 12.8 volts given directly to the DC motor because all DC motors used in the power supply device have an input

voltage of 12 Volts while 5 Volts are given to the control circuit via Arduino UNO, drivers and sensors used as voltage and current monitoring on the tennis ball throwing machine. The monitoring section is divided into two, namely current and voltage in the power circuit. The power circuit consists of a motor load source circuit that requires large power. The detail can be seen in Table 4.

TABLE 4. VOLTAGE TESTING WITH LOAD

Roller Motor Speed (RPM)	Power Circuit Voltage (Volts)
	Actual Value
3616	12.51
3616	12.51
3439	12.50
3439	12.51

5. CONCLUSION

This research successfully developed a smart tennis ball launching robot equipped with YOLO (You Only Look Once)-based intelligent control. The primary objective was to enhance training convenience and provide technological innovation for athletes. The system integrates several hardware components, including an Arduino UNO, DC motors, MG995 servos, a BTS7960 motor driver, and a camera. Based on the testing results and analysis of both the hardware and software components of the tennis ball launching machine, the following conclusions can be drawn. The ball launching machine was successfully designed and controlled using an image processing system, effectively assisting players during training sessions. The YOLO algorithm was able to accurately detect player positions within designated areas (1 to 4) and launch tennis balls accordingly with good precision. The automation system implemented in the tennis ball launcher represents a significant improvement in efficiency and functionality compared to the previous version, thereby enhancing athletes' training effectiveness, and the system is capable of operating with PWM (Pulse Width Modulation) input values of 200 and 250, achieving a maximum motor speed of 3.625 RPM (Revolutions Per Minute).

REFERENCES

- [1] P. Aversa, T. P. Moliterno, K. W. Rockmann, M. S. Bothner, D. Sharapov, and R. Eckardt, "EXPLORING NEW PLAYING FIELDS: SPORTS SETTINGS AND MANAGEMENT THEORIZING," *Acad. Manag. Discov.*, vol. 11, no. 2, pp. 145–151, 2025, doi: <https://doi.org/10.5465/amd.2025.0064>
- [2] Y. Zhu, D. Liu, Z. Gao, and Q. Zhong, "Intelligent sports interactive monitoring system based on self-driven sensing-augmented reality fusion," *Appl. Phys. Lett.*, vol. 126, no. 26, 2025, doi: <https://doi.org/10.1063/5.0276388>
- [3] Y. Zheng, L. Mo, and Z. Ye, "A Robust Multi-Dimensional n-Valued Refined Neutrosophic Logic Framework for Competitive Calculation in the Leisure Sports Industry," *Neutrosophic Sets Syst.*, vol. 86, pp. 631–643, 2025, doi: 10.5281/zenodo.15514495.
- [4] J. Padulo, "Sport and health science: Interdisciplinary approaches to modern challenges," *Br. Med. Bull.*, vol. 155, no. 1, 2025, doi: <https://doi.org/10.1093/bmb/ldaf007>
- [5] E. Glebova, M. Desbordes, and O. Czeglédi, "The 'Clockwork' Model for Deployment Technology Innovations in Sports Industry Ecosystem: Holistic Approach," *Societies*, vol. 14, no. 2, 2024, doi: <https://doi.org/10.3390/soc14020023>
- [6] Y. Zhong, "Enhancing game outcome prediction in the Chinese basketball league through a machine learning framework based on performance data," *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-08882-7>
- [7] X. Hu, Y. Mao, G. Wang, Q. Li, J. Zhang, and Y. Ji, "Catching spinning table tennis balls in simulation with end-to-end Curriculum Reinforcement Learning," *Eng. Appl. Artif. Intell.*, vol. 158, 2025, doi: <https://doi.org/10.1016/j.engappai.2025.111285>
- [8] S. Mohottala, A. Gawesha, D. Kasthurirathna, P. Samarasinghe, and C. Abhayaratne, "Spatio-temporal graph neural network based child action recognition using data-efficient methods: A systematic analysis," *Comput. Vis. Image Underst.*, vol. 259, 2025, doi: 10.1016/j.cviu.2025.104410. <https://doi.org/10.1016/j.cviu.2025.104410>
- [9] P. Pietraszewski *et al.*, "The Role of Artificial Intelligence in Sports Analytics: A Systematic Review and Meta-Analysis of Performance Trends," *Appl. Sci.*, vol. 15, no. 13, 2025, doi: <https://doi.org/10.3390/app15137254>
- [10] P. Gong and X. Luo, "A Survey of Video Action Recognition Based on Deep Learning," *Knowledge-Based Syst.*, vol. 320, 2025, doi: <https://doi.org/10.1016/j.knosys.2025.113594>
- [11] Y. Chen, J. Guo, C. Hao, and C. Song, "Multi-Source Data Fusion Method Research on the Reconstruction and Expansion Project of Long-Line Expressway," *Teh. Vjesn.*, vol. 32, no. 1, pp. 149–156, 2025, doi: <https://doi.org/10.17559/TV-20240424001492>
- [12] R. M. Sharma, "Artificial intelligence in medical image analysis and molecular diagnostics: recent advances and applications," *J. Med. Artif. Intell.*, vol. 8, 2025, doi: <https://doi.org/10.21037/jmai-24-412>
- [13] E. Shvartz, O. Zur, L. Attal, Z. Nujeidat, G. Plopsky, and D. Bahir, "GPT-4o vs. residency exams in ophthalmology—a performance analysis," *J. Med. Artif. Intell.*, vol. 8, 2025, doi: <https://doi.org/10.21037/jmai-24-269>
- [14] S. Katreddi, A. Midatani, A. P. Roy, U. Velpuri, and S. Kasani, "Pediatric pneumonia X-ray image classification: predictive model development with DenseNet-169 transfer learning," *J. Med. Artif. Intell.*, vol. 8, 2025, doi <https://doi.org/10.21037/jmai-24-356>

- [15] Z. Ding *et al.*, “Innovative design of Ag-based braze alloy and composites for dissimilar joints: Strategies and prospects,” *J. Mater. Sci. Technol.*, vol. 239, pp. 180–194, 2025, doi: <https://doi.org/10.1016/j.jmst.2025.04.006>
- [16] A. R. Singh, S. Athisayamani, F. K. Karim, A. Z. Ibrahim, S. Alshetewi, and S. M. Mostafa, “An enhanced denoising system for mammogram images using deep transformer model with fusion of local and global features,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-89451-w>
- [17] S. Prabhakaran and S. M. Praveena, “Improved gated recurrent unit-based osteosarcoma prediction on histology images: a meta-heuristic-oriented optimization concept,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-85149-1>
- [18] Q. Gong, X. Liu, and Y. Ma, “Real-Time Facial Expression Recognition Based on Image Processing in Virtual Reality,” *Int. J. Comput. Intell. Syst.*, vol. 18, no. 1, 2025, doi: <https://doi.org/10.1007/s44196-024-00729-9>
- [19] P. Pitchandi and R. K. Grace, “An optimal document clustering method using hybrid optimal feature selection and efficient soft computing classifier,” *Expert Syst. Appl.*, vol. 294, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.128762>
- [20] F. Ying, Z. Yanwei, L. Xi, Y. Peidong, and B. Miao, “Improvements in automatic diagnosis methods for knee osteoarthritis based on deep learning,” *Chinese J. Tissue Eng. Res.*, vol. 29, no. 35, pp. 7511–7518, 2025, doi: 10.12307/2026.533.
- [21] G. Chen, Y. Zhang, Z. He, and Y. Huang, “Nighttime airport runway FOD intrusive detection through frequency-domain interference of spatially aggregated dynamic feature,” *Expert Syst. Appl.*, vol. 294, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.128719>
- [22] V. Dachepalli and S. Gavini, “A Virtually assisted digital twin enabled object detection in smart industrial manufacturing,” *Expert Syst. Appl.*, vol. 294, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.128574>
- [23] G. Liu, C. Jin, Y. Ni, T. Yang, and Z. Liu, “UCIW-YOLO: Multi-category and high-precision obstacle detection model for agricultural machinery in unstructured farmland environments,” *Expert Syst. Appl.*, vol. 294, 2025, doi: <https://doi.org/10.1016/j.eswa.2025.128686>
- [24] H. Dao Viet *et al.*, “Validation of YOLOv8 algorithm in detecting colon polyps in endoscopy videos,” *J. Med. Artif. Intell.*, vol. 8, 2025, doi: <https://doi.org/10.21037/jmai-24-338>
- [25] H. Tong, D. Zhu, H. Guo, and Z. Zhang, “Adaptive GLMB filter with IoU-based birth modeling for UAV visual multi-object tracking,” *Signal Processing*, vol. 237, 2025, doi: <https://doi.org/10.1016/j.sigpro.2025.110087>
- [26] Q. Li and W. Chen, “Lightweight vision transformer model for pine wilt disease detection using aerial RGB image and adversarial data augmentation,” *Smart Agric. Technol.*, vol. 12, 2025, doi: <https://doi.org/10.1016/j.atech.2025.101167>
- [27] L. Qiao, M. Veltrup, and B. Mayer, “Comparison of mask R-CNN and YOLOv8-seg for improved monitoring of the PCB surface during laser cleaning,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-02131-7>
- [28] G. Liu *et al.*, “A weakly-supervised follicle segmentation method in ultrasound images,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-95957-0>
- [29] W. Mansouri, M. A. Alohal, H. Alqahtani, N. Alruwais, M. Alshammeri, and A. Mahmud, “Deep convolutional neural network-based enhanced crowd density monitoring for intelligent urban planning on smart cities,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-90430-4>
- [30] Y. Li, X. Li, and M. Lin, “FE-YOLO: Fourier enhancement YOLO for end-to-end object detection in low-light conditions,” *Digit. Signal Process. A Rev. J.*, vol. 166, 2025, doi: <https://doi.org/10.1016/j.dsp.2025.105355>
- [31] D. L. Tukeev, O. V. Afanaseva, and T. F. Tulyakov, “Realization of Statistical Models Based on Symmetric Unimodal Distributions,” *Int. J. Eng. Trans. B Appl.*, vol. 39, no. 6, pp. 407–419, 2026, doi: <https://doi.org/10.5829/ije.2026.39.02b.10>
- [32] O. O. Okundalaye, N. Özdemir, and R. O. Awonusika, “Early breast cancer prediction using optimized machine learning and tumor-immune modeling,” *J. Comput. Appl. Math.*, vol. 473, 2026, doi: <https://doi.org/10.1016/j.cam.2025.116875>
- [33] M. Braun and O. Ezenwachukwu, “Three dimensional calculations of the ground state energy for the electronic wave function of the Coulomb two centre problem using basis functions related to the sinc function,” *J. Comput. Appl. Math.*, vol. 473, 2026, doi: <https://doi.org/10.1016/j.cam.2025.116779>
- [34] M. Bastam, D. Hashem Majeed, and M. Babagoli, “A Hybrid Ensemble Framework for Smart Irrigation: Optimizing Water Management in Precision Agriculture,” *Int. J. Eng. Trans. B Appl.*, vol. 39, no. 3, pp. 705–726, 2026, doi: <https://doi.org/10.5829/ije.2026.39.03c.12>
- [35] A. Mahdi and A. Al Saadi, “Deep Learning- based Smart Traffic Prediction for Enhanced Quality of Service in Software-defined Networking,” *Int. J. Eng. Trans. B Appl.*, vol. 39, no. 3, pp. 739–755, 2026, doi: <https://doi.org/10.5829/ije.2026.39.03c.14>
- [36] H. M. U. Ali, Q. Farooq, A. Imran, and K. El Hindi, “A systematic literature review on sentiment analysis techniques, challenges, and future trends,” *Knowl. Inf. Syst.*, vol. 67, no. 5, pp. 3967–4034, 2025, doi: <https://doi.org/10.1007/s10115-025-02365-x>
- [37] A. Kabir, S. Wang, Y. Tian, T.-H. Chen, M. Asaduzzaman, and W. Zhang, “ZS4C: Zero-Shot Synthesis of Compilable Code for Incomplete Code Snippets Using LLMs,” *ACM Trans. Softw. Eng. Methodol.*, vol. 34, no. 4, 2025, doi: <https://doi.org/10.1145/3702979>
- [38] B. Casey, J. C. S. Santos, and G. Perry, “A Survey of Source Code Representations for Machine Learning-Based Cybersecurity Tasks,” *ACM Comput. Surv.*, vol. 57, no. 8, 2025, doi: <https://doi.org/10.1145/3721977>
- [39] J. S. Notland, M. Nowostawski, and J. Li, “An Empirical Study on Governance in Bitcoin’s Consensus Evolution,” *ACM Trans. Softw. Eng. Methodol.*, vol. 34, no. 3, 2025, doi: <https://doi.org/10.1145/3699600>
- [40] D. Palla and A. Slaby, “Evaluation of Generative AI Models in Python Code Generation: A Comparative Study,” *IEEE Access*, vol. 13, pp. 65334–65347, 2025, doi: <https://doi.org/10.1109/ACCESS.2025.3560244>
- [41] S. Pastore *et al.*, “ClearFinder: a Python GUI for annotating cells in cleared mouse brain,” *BMC Bioinformatics*, vol. 26, no. 1, 2025, doi: <https://doi.org/10.1186/s12859-025-06039-x>
- [42] D. Swanlund and N. Schuurman, “MaskMyPy: python tools for performing and analyzing geographic masks,” *Int. J. Health Geogr.*, vol. 24, no. 1, 2025, doi: <https://doi.org/10.1186/s12942-025-00399-6>
- [43] A. Ghimire, Y. S. Chung, S. Jeong, and Y. Kim, “Python algorithm package for automated Estimation of major legume root traits using two dimensional images,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-91993-y>
- [44] H. Qin, M. Hussain, M. F. Hanif, M. K. Siddiqui, Z. Hussain, and M. A. Fiidow, “On QSPR analysis of pulmonary cancer drugs using python-driven topological modeling,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-88419-0>
- [45] P. M. A. Sadique and R. V. Aswiga, “Automatic summarization of cooking videos using transfer learning and transformer-based models,” *Discov. Artif. Intell.*, vol. 5, no. 1, p. 7, 2025, doi: <https://doi.org/10.1007/s44163-025-00230-y>
- [46] N. H. Sabbry and A. Levina, “An optimized elliptic curve digital signature strategy for resource-constrained devices,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-05601-0>

- [47] H. Guo *et al.*, “Global trends and evolution of aquavoltaics in sustainable aquaculture and energy generation,” *Renew. Energy*, vol. 255, 2025, doi: <https://doi.org/10.1016/j.renene.2025.123864>
- [48] V.-F. Drăgoi, B. Colombier, N. Vallet, P.-L. Cayrel, and V. Grosso, “Full Key-Recovery Cubic-Time Template Attack on Classic McEliece Decapsulation,” *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2025, no. 1, pp. 367–391, 2025, doi: <https://doi.org/10.46586/tches.v2025.i1.367-391>
- [49] A. Kurian, A. Dubey, F. Yaman, and A. Aysu, “TPUXtract: An Exhaustive Hyperparameter Extraction Framework,” *IACR Trans. Cryptogr. Hardw. Embed. Syst.*, vol. 2025, no. 1, pp. 78–103, 2025, doi: <https://doi.org/10.46586/tches.v2025.i1.78-103>
- [50] T. King, Y. Zhou, T. Röddiger, and M. Beigl, “MicroNAS for memory and latency constrained hardware aware neural architecture search in time series classification on microcontrollers,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-90764-z>
- [51] Y. Lu, T. Huang, J. Zhang, X. Wang, and X. Guo, “Adaptive finite-time fuzzy prescribed performance fault-tolerant control for uncertain active suspensions with actuator nonlinear characteristics,” *J. Sound Vib.*, vol. 618, 2025, doi: <https://doi.org/10.1016/j.jsv.2025.119232>
- [52] R. Asnawi *et al.*, “Fuzzy Logic Controller-Proportional-Integral for Motor Velocity Control of Electric Rail Train Using DC-DC Converter,” *J. Adv. Res. Appl. Sci. Eng. Technol.*, vol. 54, no. 1, pp. 62–79, 2025, doi: <https://doi.org/10.37934/araset.54.1.6279>
- [53] M. A. Bhayo *et al.*, “High precision experimentally validated adaptive neuro fuzzy inference system controller for DC motor drive system,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-97549-4>
- [54] X. Li, Y. Zhang, and J. Wu, “Optimization of tangential interior brushless DC motor rotor for hybrid vehicles,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-98799-y>
- [55] M. A. Khan, D.-E.-Z. Baig, H. Ali, and F. R. Albogamy, “Optimized System Identification (SI) of Brushless DC (BLDC) motor using Data-Driven Modeling Methods,” *Sci. Rep.*, vol. 15, no. 1, 2025, doi: <https://doi.org/10.1038/s41598-025-93444-0>
- [56] B. Meenakshi, J. C. Lavanya, and R. Sivaprasad, “Versa Watt Pioneering a Multi-Output SMPS Solution for Next-Gen Devices,” in *2025 International Conference on Computing and Communication Technologies (ICCCCT)*, IEEE, 2025, pp. 1–4. <https://doi.org/10.1109/ICCCCT63501.2025.11019084>
- [57] F. Hashemzadeh, V. Mohammadi, and M. Sabahi, “Hybrid Control of Bidirectional DC-DC Power Source of Electric Vehicles With Dynamical Brake,” in *2025 IEEE 19th International Conference on Compatibility, Power Electronics and Power Engineering (CPE-POWERENG)*, IEEE, 2025, pp. 1–6. <https://doi.org/10.1109/CPE-POWERENG63314.2025.11027231>
- [58] S. Kumar, S. Mondal, A. B. Choudhury, H. S. Bhattacharyya, and C. K. Chanda, “Estimation of state of charge for a lithium-ion battery pack using nonlinear Kalman filters,” *Ionics (Kiel)*, pp. 1–16, 2025. <https://doi.org/10.1007/s11581-025-06420-z>
- [59] S. Nau, J. Krummenauer, and A. Zimmermann, “SPICEAssistant: LLM using SPICE Simulation Tools for Schematic Design of Switched-Mode Power Supplies,” *arXiv Prepr. arXiv2507.10639*, 2025.
- [60] M.-S. Kim, J.-W. Sung, Y.-H. Kim, S.-S. Kwak, and S.-D. Choi, “Dual-Band-Based Moving Z-Score Algorithm for Detecting Series AC Arc Faults,” *J. Electr. Eng. Technol.*, pp. 1–13, 2025. <https://doi.org/10.1007/s42835-025-02203-8>