

Design and Construction of an Electric Wheelchair with Joystick-Based DC Motor Control and YOLO Object Detection

Intan Permata Agustin

Department Of Electrical Engineering, Faculty Of Vocational Studies, Universitas Negeri Surabaya, Indonesia
e-mail address: intan.20011@mhs.unesa.ac.id (corresponding author)

Received: 15 September 2025 | Revised: 15 October 2025 | Accepted: 30 October 2025

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



ABSTRACT

Electric wheelchairs have become an essential solution for individuals with mobility limitations; however, safe and responsive control remains a primary challenge. This thesis aims to design an electric wheelchair system integrated with joystick control and object detection technology using YOLOv4 to enhance safety, accuracy, and user-friendliness. The joystick control system serves as the primary user interface for wheelchair navigation, while the safety system utilizes computer vision technology based on YOLOv4. The computational process is executed on a Raspberry Pi 3 device. The electric wheelchair automatically stops and provides audio warnings when obstacles are detected, implemented using an Arduino Uno to control the speed and direction of DC motors while ensuring user safety. An experimental method is employed in the design of this electric wheelchair. Testing results demonstrate that the joystick-based control system responds appropriately to user commands, including forward, backward, right, left, rotation, and braking. The YOLO object detection system achieves an average object detection accuracy of 98% with an average computational time of 3 to 5 seconds. The wheelchair can support a maximum load of 70 kg.

Keywords: *Electric wheelchair; Joystick control; YOLOv4; Object detection; Computer vision.*

1. INTRODUCTION

Based on data from the National Economic Survey in 2020, approximately 28.5 million Indonesians experience disabilities. Wheelchairs serve as essential assistive devices commonly used to support mobility for individuals with motor system impairments in their legs. While manual wheelchairs are widely utilized, they present several limitations, including high energy requirements and dependence on assistance from others for daily activities.

As a solution to these challenges, electric wheelchairs have emerged as a more efficient alternative. Electric wheelchairs offer advantages in terms of power and control, providing easier movement for users, particularly those who navigate independently[1][2][3][4][5]. With technological advancements, various innovations have been implemented in electric wheelchairs, including joystick-based control systems. However, there remains potential to enhance electric wheelchair functionality through the integration of computer vision technology.

One technology that can be integrated is YOLO (You Only Look Once), a fast and efficient object detection method. YOLO employs a simple detection model to recognize objects in images by applying single regression, creating bounding boxes with associated class probabilities[6][7][8][9][10]. Integrating YOLO into electric wheelchairs enables real-time object detection around the wheelchair, potentially enhancing user safety and comfort.

The integration of Arduino IDE (Integrated Development Environment) as a microcontroller facilitates wheelchair control by generating PWM signal outputs transmitted to motor drivers, enabling speed control of electric wheelchair motors. This allows the electric wheelchair to adjust its speed or take corrective actions based on surrounding conditions. In addition to braking mechanisms, audio warnings are also provided when approaching obstacles. Therefore, this research aims to integrate joystick control with object detection using YOLOv4. Through this technological combination, it is expected that wheelchair user mobility and independence can be significantly enhanced.

Several studies on electric wheelchairs have been conducted using various methods. For instance, the Guided Following Control System on Smart Wheelchairs using the YOLOv5 method based on Nvidia Jetson TX2 demonstrates that by employing

YOLOv5 to detect and track predetermined guides, smart electric wheelchairs can follow guides smoothly and responsively, enabling users to navigate environments easily.

Joystick is a computer input device featuring a lever that can be moved in various directions along with multiple buttons around it to perform additional functions [11][12][13][14]. Lever movements are converted into electronic signals, which are then transmitted to the device being used. The joystick can continuously detect movement changes on each axis and send corresponding analog signals. YOLO is an algorithm developed for real-time object detection. This detection system reuses classifiers or locators to identify objects. Images are divided into grid cells with sizes adjusted based on input dimensions. YOLO's core advantage is its high-speed detection without significantly compromising accuracy. This method can perform real-time object detection with optimal performance, making it suitable for various applications requiring quick responses. This research contributes to the field of assistive technology by developing an integrated electric wheelchair system that combines:

- Joystick-based user interface for intuitive navigation control
- YOLOv4 object detection system for real-time obstacle identification and safety enhancement
- Arduino Uno microcontroller integration for efficient motor speed and direction control
- Audio warning system for improved user awareness and safety
- Comprehensive system architecture that demonstrates the feasibility of combining computer vision technology with traditional wheelchair control systems

The integration of these technologies aims to significantly improve wheelchair user mobility, independence, and safety in daily navigation tasks.

2. METHOD

2.1. Raspberry Pi 3

Raspberry Pi 3 is a single-board computer (SBC) developed by the Raspberry Pi Foundation. Released in February 2016, it represents the third generation of the popular Raspberry Pi series and serves as a compact, affordable computing platform for various applications including education, prototyping, and embedded systems [15][16][17][18][19]. Advantages for Projects Like Electric Wheelchair:

- Integrated Connectivity: Built-in Wi-Fi and Bluetooth eliminate need for external adapters
- Processing Power: Quad-core processor sufficient for computer vision tasks like YOLOv4
- GPIO Capabilities: 40-pin header allows connection to various sensors and actuators
- Camera Support: Direct CSI camera interface for real-time image processing
- Cost-Effective: Affordable price point makes it accessible for research and development
- Community Support: Extensive documentation and community resources

Limitations:

- Processing Constraints: Limited computational power compared to high-end computers
- Memory Limitation: 1GB RAM may be insufficient for complex deep learning models
- Thermal Management: May require heat sinks for intensive computational tasks
- Power Consumption: Higher power consumption compared to microcontrollers

The Raspberry Pi 3's balance of processing capability, integrated features, and cost-effectiveness makes it well-suited for this type of embedded assistive technology application, particularly when real-time computer vision processing is required alongside traditional control systems.

2.2. Arduino Uno

The Arduino Uno is an ATmega328P-based microcontroller development board used to create and control electronic systems such as sensors, motors, and other smart devices [20][21][22][23][24]. Arduino is an open-source platform for creating programmable electronic projects. It provides easy-to-use hardware (such as the Arduino Uno) and software (the Arduino IDE). The Arduino Uno is the most common and standard Arduino series, suitable for both beginners and professionals. The name "Uno" means "one" in Italian, signifying the first stable version of the Arduino platform. Arduino Uno Functions:

- Reads input from sensors (e.g., temperature, light, distance).
- Controls outputs, such as turning on LEDs, driving motors, or sending data to a computer.
- Automates systems, such as automatic alarms, robots, irrigation systems, etc.
- Serial communication, such as sending data to a computer or to other devices via USB, Bluetooth, WiFi, or LoRa.

Main Components of the Arduino Uno:

- ATmega328P Microcontroller
- 14 digital I/O pins
- 6 analog input pins

- USB port for programming
- Voltage regulator
- 16 MHz crystal oscillator

Arduino Uno Advantages:

- Easy to program with the Arduino IDE
- Numerous examples and documentation
- Compatible with many sensors and modules
- Suitable for learning and prototyping

2.3. DC Motor

A DC motor (Direct Current Motor) is an electric motor that converts direct current (DC) electrical energy into mechanical energy in the form of rotational motion. When DC current flows through a coil within a magnetic field, an electromagnetic force (Lorentz's Law) is generated that rotates the rotor (the rotating part of the motor) [25][26][27][28][29][30]. Table 1 outlines the fundamental components of a DC (Direct Current) motor and their respective functions. Understanding these components is essential for comprehending how DC motors operate in applications such as electric wheelchairs, where reliable and controlled mechanical motion is critical for user mobility and safety.

TABLE 1. MAIN COMPONENTS OF A DC MOTOR

Components	Functions
Rotor (Armature)	Rotating part, containing coils or coils of wire
Stator	Fixed part that provides the magnetic field
Brush	Provides electrical connection to the commutator and armature
Commutator	Changes the direction of current in the coils to maintain the direction of rotation of the rotor
Shaft	Holder for wheels, fans, or other mechanical loads

Important Characteristics of a DC Motor:

- Operating voltage (e.g., 3V, 6V, 12V)
- Rotation speed (RPM – Rotations Per Minute)
- Torque (twisting force)
- Direction of rotation (can be changed by reversing the voltage polarity)

2.4. YOLO

YOLO (You Only Look Once) is a deep learning-based object detection method that can recognize and assign bounding boxes to various objects in an image in one pass. YOLO is known for its speed and efficiency in processing images in real-time, suitable for applications such as smart cameras, autonomous vehicles, and robotics. YOLO uses a Convolutional Neural Network (CNN) architecture, where images are processed gradually through convolutional layers to extract features, then classified and predicted directly [31][32][33][34][35]. Table 2 compares YOLO (You Only Look Once) with other popular object detection methods, particularly R-CNN (Region-based Convolutional Neural Networks) and SSD (Single Shot Detector). The comparison highlights the fundamental differences in approach, speed, and efficiency between these object detection algorithms.

TABLE 2. YOLO VS OTHER METHODS:

Other Methods (R-CNN, SSD)	YOLO
Multi-stage	One-step
Slow (many processes)	Fast (real-time)
Accurate detection but heavy	Fast and efficient detection

Table 3 presents the evolution of YOLO (You Only Look Once) object detection algorithms, highlighting the key improvements and distinctive features of each version. The progression shows how the YOLO family has continuously evolved to balance accuracy, speed, and usability while expanding its capabilities beyond basic object detection

TABLE 3. YOLO VS OTHER METHODS:

Version	Distinctive Features
YOLOv1	First version: real-time but less precise
YOLOv3	More accurate and faster
YOLOv4	Optimized for accuracy and speed
YOLOv5	Written in PyTorch, easier to implement
YOLOv8	Latest version: supports classification, segmentation, and tracking

Examples of YOLO Applications:

- Smart security cameras (person or vehicle detection)
- Autonomous cars (sign and pedestrian detection)
- Tennis or soccer robots (ball and player detection)
- Road damage detection
- Face mask detection

YOLO Advantages [36][37][38][39][40]:

- Real-time (very fast)
- End-to-end learning (directly from image to detection output)
- Multitask: Can detect multiple objects simultaneously

3. PROPOSED METHOD

This paper uses an experimental method to design a joystick-based electric wheelchair control system with YOLOv4 detection for speed control using the Arduino IDE (Integrated Development Environment) in the electric wheelchair control system. The hardware design was implemented by implementing the schematic design in the wheelchair. The circuit schematic is presented in Figure 1 of the hardware design. In the hardware design, wheelchair control is implemented using a joystick connected to an Arduino ATmega328 microcontroller. The Arduino processes signals from the joystick to control wheelchair movements such as forward, backward, turning left, turning right, and stopping. The PWM signal from the Arduino is forwarded to the IBT02 motor driver, which controls the power, speed, and direction of the DC motor. The motor is powered by two 12V 12AH batteries connected in parallel, producing a voltage of 24V. Additionally, the wheelchair is equipped with a USB camera connected to a Raspberry Pi3 running YOLOv4 to detect obstacles. The Raspberry Pi3 sends data to the Arduino via a serial connection, which adjusts the speed and stops the motors, and provides an alert if there is an obstacle in front or behind the wheelchair. The following is the design result of the device.

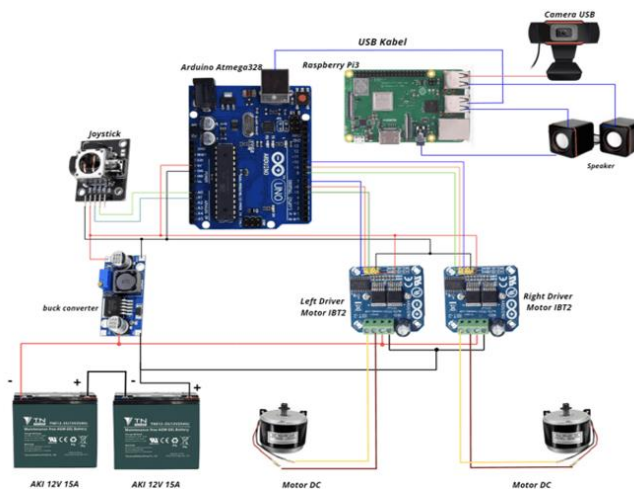


Figure 1. Tool Design

The software design was carried out by implementing the schematic design for the wheelchair. The circuit schematic is shown in Figure 2. Software design. Connect a USB camera to the Raspberry Pi3, install YOLOv4, and configure it to detect objects in front of and behind the wheelchair. Connect the Raspberry Pi3 to the Arduino via a serial connection to control the motor driver, which reduces speed and provides warnings when obstacles are encountered. Connect a joystick to the Arduino to control the wheelchair with signals for forward, backward, left turn, right turn, and stop. Program the Arduino to control the DC motor based on instructions from the Raspberry Pi3 and joystick.

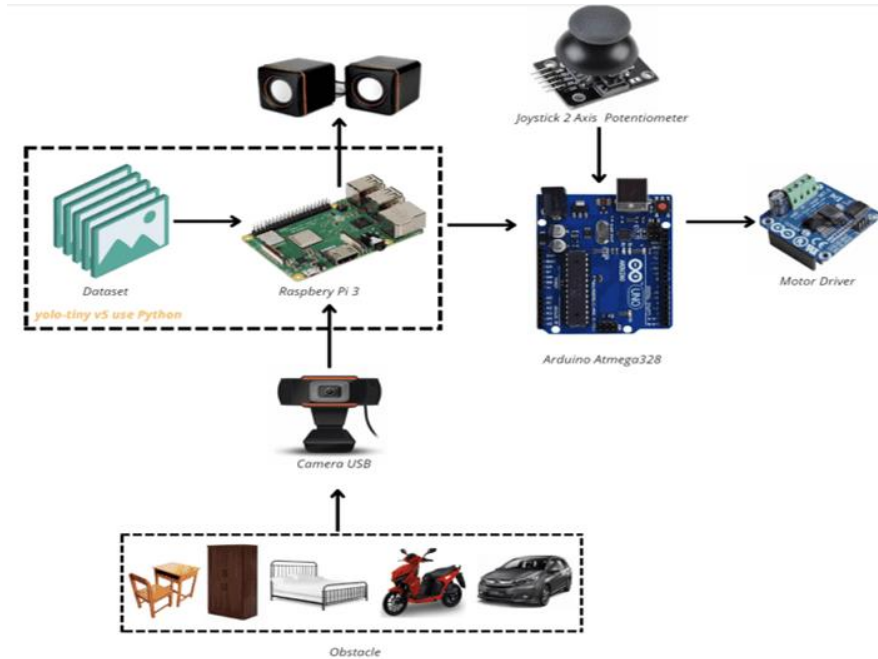


Figure. 2. Software Design

4. RESULT AND DISCUSSION

To determine the performance of the proposed tool, this article conducted 4 case studies, namely Joystick Control System Testing, Yolo Detection, and Automatic Braking Testing on Electric Wheelchairs.

4.1. Joystick Control System Testing

The purpose of testing the joystick control system is to evaluate the extent to which the commands given correspond to the desired direction when the joystick lever is moved. Based on the test results, the movement of the joystick lever on the wheelchair indicates that the motor rotates according to the input provided by the joystick.

TABLE 4. JOYSTICK TEST

No	Joystick Control System	Right DC Motor	Left DC Motor
1	Forward	V	V
2	Backward	V	V
3	Right	-	V
4	Left	V	-

Based on Table 4, the joystick control system test results show that the voltage (V) values indicate the motors rotate according to the commanded directions. The test was successful as the motors responded accurately to joystick inputs. When the joystick is directed forward, both motors (right and left) rotate clockwise. Conversely, when the joystick is directed backward, both motors rotate counterclockwise. For rightward movement, the left motor rotates at full speed while the right motor operates at minimum speed, and vice versa for leftward movement.

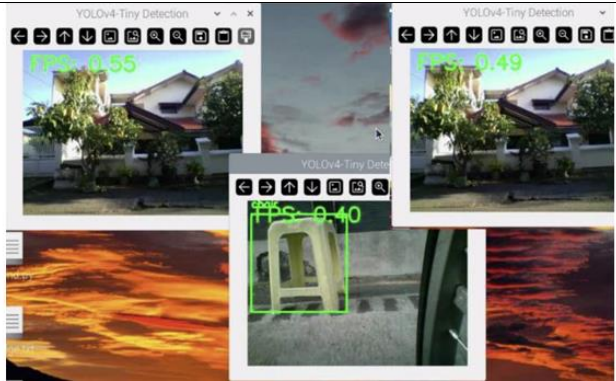

4.2. Yolo Detection Testing

This YOLOv4 control system test aims to assess the detection accuracy and computation time of YOLOv4 when detecting obstacles in front of and behind the wheelchair. The purpose of this YOLOv4 control system test was to evaluate the detection accuracy and computational time of YOLOv4 when detecting obstacles in front of and behind the wheelchair. The test results showed that the lowest detection accuracy was 90%, while the highest detection accuracy reached 100%, with an overall average accuracy of 98%. However, the system encountered limitations due to the Raspberry Pi 3 specifications, resulting in an average computational time of 3-5 seconds. The Data can be seen in Table 5.

TABLE 5. YOLO DETECTION SYSTEM

No	Objects	Akurasi (%)	Average computation time (s)	Distance (cm)
1	Chairs	100%	4	50
2	Tables	100%	5	50
3	Motorcycles	100%	3	50
4	Cars	100%	3	50
5	Cupboards	100%	5	50
6	Beds	100%	5	50

TABLE 6. AUTOMATIC BRAKING TESTING

No	Detection	Information
1		The rear camera detects an object at a distance of 500 cm, causing the wheelchair to stop and sound a warning sound.
2		Objects are detected by the front and rear cameras at a distance of 1 m and 500 cm, respectively, so the wheelchair stops and emits a warning sound.

4.3. Automatic Braking Testing on Electric Wheelchairs Testing

Automatic braking tests were conducted to ensure the device and program functioned as expected. Table 3 shows that the system's safety level was quite high, as the wheel movement matched the object tracking. The wheelchair's movement was as expected; when it approached an object, it automatically stopped and emitted a warning sound. Automatic braking tests were conducted to ensure the device and program functioned as expected. Table 6 shows that, based on the two experiments, the system's safety level is quite high because the wheel movement matches the object tracking. However, there were some minor

errors due to object similarity. These did not impede the wheelchair's function or movement. When an obstacle was detected, the wheel movement stopped as expected.

5. CONCLUSION

Electric wheelchairs have become essential for individuals with mobility limitations; however, safe and responsive control remains challenging. This thesis presents an electric wheelchair system integrating joystick control with YOLOv4 object detection technology to enhance safety, accuracy, and user-friendliness. The joystick serves as the primary navigation interface, while YOLOv4-based computer vision provides obstacle detection. The system utilizes a Raspberry Pi 3 for computation and Arduino Uno for motor control. The experimental design demonstrates that the joystick-based control system effectively responds to user commands including forward, backward, right, left, rotation, and braking movements. Testing results reveal the YOLOv4 system achieves 90-100% detection accuracy with an average of 98%, processing obstacles within 3-5 seconds. The wheelchair automatically stops and provides audio warnings when obstacles are detected within 50 cm to 1 meter range. Key findings include: (1) Successful integration of joystick control and YOLOv4 technology significantly enhances user safety; (2) The integrated system effectively regulates DC motor speed and adapts to surrounding obstacles; (3) Automatic braking mechanism functions effectively, detecting objects and stopping the wheelchair when obstacles are within 50 cm proximity; (4) The combined joystick-YOLOv4 system demonstrates excellent responsiveness and safety performance in both control and object detection capabilities. This research contributes to assistive technology by developing an affordable, intelligent wheelchair system that combines intuitive user control with advanced safety features, potentially improving mobility and independence for users with disabilities.

REFERENCE

- [1] W. Szaj, M. Wanic, W. Wojnarowska, and S. Miechowicz, "Mechatronic Anticollision System for Electric Wheelchairs Based on a Time-of-Flight Sensor," *Electronics*, vol. 14, no. 11, p. 2307, 2025. <https://doi.org/10.3390/electronics14112307>
- [2] M. Kmich, H. Karmouni, and M. Sayyouri, "Improved PID Controller using Archimedes' Optimization Algorithm to Control an Electric Wheelchair," in *2025 7th Global Power, Energy and Communication Conference (GPECOM)*, IEEE, 2025, pp. 176–181. <https://doi.org/10.1109/GPECOM65896.2025.11062018>.
- [3] Å. M.-B. Vassbø and G. Mørk, "Children's opportunities and rights for mobility and participation: Occupational therapists' experiences with paediatric powered mobility," *Scand. J. Occup. Ther.*, vol. 32, no. 1, p. 2520863, 2025. <https://doi.org/10.1080/11038128.2025.2520863>
- [4] J.-M. Ahn, K.-H. Ha, J.-H. Cho, H. Seo, and D.-K. Lim, "Optimal Design of SPMSM Using a Subdivided Meta-Model Assisted Multi-Objective Optimization Algorithm," *J. Electr. Eng. Technol.*, pp. 1–17, 2025. <https://doi.org/10.1007/s42835-025-02326-y>
- [5] A. Haddoun, D. Djabri, M. Saidani, and M. Benbouzid, "Development and Evaluation of a Head-Controlled Wheelchair System for Users with Severe Motor Impairments," *MethodsX*, p. 103485, 2025. <https://doi.org/10.1016/j.mex.2025.103485>
- [6] L. Yang and G.-Y. Chen, "Evaluation of Deep Learning for Caries Detection With Fine-Grained Classification and Postprocessing Improvements," *Int. Dent. J.*, vol. 75, no. 5, p. 100898, 2025. <https://doi.org/10.1016/j.identj.2025.100898>
- [7] Y. Liang *et al.*, "MAH-YOLO: an enhanced YOLOv8n framework for loess landslide detection with multi-attention mechanisms," *Int. J. Digit. Earth*, vol. 18, no. 1, p. 2536666, 2025. <https://doi.org/10.1080/17538947.2025.2536666>
- [8] J. Liu, C. Feng, G. Jin, and J. Fan, "EdLLM: A Novel Entity Detection Method for Test Data Integrating YOLO-World and LLM," in *International Conference on Intelligent Computing*, Springer, 2025, pp. 195–206. https://doi.org/10.1007/978-981-96-9815-8_17
- [9] Q. T. Lam, M. H. N. Le, I.-T. Lee, and N. Q. K. Le, "Evaluating YOLO for dental caries diagnosis: a systematic review and meta-analysis," *Evid. Based. Dent.*, pp. 1–14, 2025. <https://doi.org/10.1038/s41432-025-01180-1>
- [10] Q. Liu, X. Wang, A. Yang, Z. Hu, and X. Zhang, "ODC-YOLO: an optimized YOLOv5 method for detecting objects in remote sensing images," *Remote Sens. Lett.*, vol. 16, no. 10, pp. 1120–1130, 2025. <https://doi.org/10.1080/2150704X.2025.2529599>
- [11] D. Zielasko, "A Technical Overview on Cutting-Edge Human-Joystick Locomotion aka Leaning in VR: Intuitive, Fun, Calibration-Free, and Open Source," in *2025 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, IEEE, 2025, pp. 261–265. <https://doi.org/10.1109/VRW66409.2025.00062>
- [12] Y. Lee, K. A. Hull, and J. Kim, "Beyond the joystick: Exploring the relationship between game addiction trajectories and aggression dimensions," *J. Crim. Justice*, vol. 98, p. 102425, 2025. <https://doi.org/10.1016/j.jcrimjus.2025.102425>
- [13] T. Ni, Z. Zhao, P. Zhang, Y. Zhao, K. Yang, and J. Liang, "A novel estimation strategy for the operating force at the end of joystick based on parameters identification," *Intell. Serv. Robot.*, pp. 1–14, 2025. <https://doi.org/10.1007/s11370-025-00597-y>
- [14] M. Larbi, A. Maldi, E.-H. Guechi, and K. Belharet, "Extended state observer based-optimal control of a microrobot navigation along a trajectory generated by a joystick device," *IEEE Trans. Autom. Sci. Eng.*, 2025. <https://doi.org/10.1109/TASE.2025.3566475>
- [15] A. Joice *et al.*, "Applications of Raspberry Pi for Precision Agriculture—A Systematic Review," *Agriculture*, vol. 15, no. 3, p. 227, 2025. <https://doi.org/10.3390/agriculture15030227>
- [16] R. Biswas, A. Pandey, and S. Murugan, "Greenhouse Environment Monitoring Using Raspberry Pi," in *2025 International Conference on Visual Analytics and Data Visualization (ICVADV)*, IEEE, 2025, pp. 549–554. <https://doi.org/10.1109/ICVADV63329.2025.10961617>
- [17] S. Pecolt, A. Błażejowski, T. Krolkowski, I. Maciejewski, K. Gierula, and S. Glowinski, "Personal Identification Using Embedded Raspberry Pi-Based Face Recognition Systems," *Appl. Sci.*, vol. 15, no. 2, p. 887, 2025. <https://doi.org/10.3390/app15020887>
- [18] M. B. Musthafa, S. Huda, T. T. Nguyen, Y. Kodera, and Y. Nogami, "Optimized Ensemble Deep Learning for Real-Time Intrusion

- Detection on Resource-Constrained Raspberry Pi Devices,” *IEEE Access*, 2025. <https://doi.org/10.1109/ACCESS.2025.3584373>
- [19] N. Portillo, “Design and Implementation of an IoT Cluster with Raspberry Pi Powered by Solar Energy: A Theoretical Approach,” *arXiv Prepr. arXiv2503.03618*, 2025.
- [20] S. Tselegkaridis, T. Sapounidis, C. Tokatlidis, S. Rapti, and D. Papakostas, “AABC: A Tool for Assessing Arduino Basic Coding Skills,” *Comput. Appl. Eng. Educ.*, vol. 33, no. 3, p. e70029, 2025. <https://doi.org/10.1002/cae.70029>
- [21] J. A. Ramos-Hernanz, D. Teso-Fz-Betoño, I. Aramendia, M. Erauzquin, E. Kurt, and J. M. Lopez-Guede, “Smart Low-Cost On-Board Charger for Electric Vehicles Using Arduino-Based Control,” *Energies*, vol. 18, no. 8, p. 1910, 2025. <https://doi.org/10.3390/en18081910>
- [22] A. Shrivastava, S. J. Suji Prasad, A. R. Yeruva, P. Mani, P. Nagpal, and A. Chaturvedi, “IoT based RFID attendance monitoring system of students using Arduino ESP8266 & Adafruit. io on defined area,” *Cybern. Syst.*, vol. 56, no. 1, pp. 21–32, 2025. <https://doi.org/10.1080/01969722.2023.2166243>
- [23] M. Sharipov *et al.*, “Recent progress in Arduino-and smartphone-based sensors for biochemical and environmental analysis,” *TrAC Trends Anal. Chem.*, vol. 183, p. 118103, 2025. <https://doi.org/10.1016/j.trac.2024.118103>
- [24] N. T. Tsebesebe, K. Mpofo, S. Sivarasu, and P. Mthunzi-Kufa, “Arduino-based devices in healthcare and environmental monitoring,” *Discov. Internet Things*, vol. 5, no. 1, pp. 1–31, 2025. <https://doi.org/10.1002/oca.3218>
- [25] A. F. Güven, O. Ö. Mengi, M. A. Elseify, and S. Kamel, “Comprehensive Optimization of PID Controller Parameters for DC Motor Speed Management Using a Modified Jellyfish Search Algorithm,” *Optim. Control Appl. Methods*, vol. 46, no. 1, pp. 320–342, 2025, doi: 10.1002/oca.3218. <https://doi.org/10.1002/oca.3218>
- [26] A. Sen, B. Singh, K. Mahtani, A. Moradzadeh, and S. M. Muyeen, “Optimized design of a permanent magnet brushless DC motor for solar water-pumping applications,” *Results Eng.*, vol. 26, 2025, doi: 10.1016/j.rineng.2025.104633.
- [27] E. Arévalo, R. Herrera Hernández, D. Katselis, C. Reusser, and R. Carvajal, “On Modelling and State Estimation of DC Motors,” in *Actuators*, MDPI, 2025, p. 160. <https://doi.org/10.3390/act14040160>
- [28] S. Ekinci *et al.*, “Advanced control parameter optimization in DC motors and liquid level systems,” *Sci. Rep.*, vol. 15, no. 1, p. 1394, 2025. <https://doi.org/10.1038/s41598-025-85273-y>
- [29] M. H. Setiawan, A. Ma’arif, M. F. Saifuddin, and W. A. Salah, “A Comparative Study of PID, FOPID, ISF, SMC, and FLC Controllers for DC Motor Speed Control with Particle Swarm Optimization,” *Int. J. Robot. Control Syst.*, vol. 5, no. 1, pp. 640–660, 2025. <https://doi.org/10.31763/ijrcs.v5i1.1764>
- [30] X. Zheng, H. Wen, X. Yang, X. Yu, and J. J. Rodriguez-Andina, “Adaptive neural zeta-backstepping with predefined damping ratio. Application to DC motors,” *IEEE Trans. Cybern.*, 2025. <https://doi.org/10.1109/TCYB.2025.3539544>
- [31] T.-H. Tsai, N.-C. Tung, and C.-Y. Chen, “An FPGA-Based Reconfigurable Convolutional Neural Network Accelerator for Tiny YOLO-V3,” *Circuits, Syst. Signal Process.*, vol. 44, no. 5, pp. 3388–3409, 2025. <https://doi.org/10.1007/s00034-024-02968-9>
- [32] S. Pericherla and E. Ilavarasan, “Cyberbullying detection and classification on social media images using Convolution Neural Networks and CB-YOLO model,” *Evol. Syst.*, vol. 16, no. 2, p. 43, 2025. <https://doi.org/10.1007/s12530-025-09656-2>
- [33] R. Raushan, V. Singhal, and R. K. Jha, “Damage detection in concrete structures with multi-feature backgrounds using the YOLO network family,” *Autom. Constr.*, vol. 170, p. 105887, 2025. <https://doi.org/10.1016/j.autcon.2024.105887>
- [34] R. Ayachi, Y. Said, M. Afif, A. Alshammari, M. Hleili, and A. Ben Abdelali, “Assessing YOLO models for real-time object detection in urban environments for advanced driver-assistance systems (ADAS),” *Alexandria Eng. J.*, vol. 123, pp. 530–549, 2025. <https://doi.org/10.1016/j.aej.2025.03.077>
- [35] S. Fu, H. Pan, J. Huang, X. Zhang, and Z. Jing, “AGD-YOLO: A forward-looking sonar target detection method with attention-guided denoising convolutional neural network,” *Aerosp. Syst.*, pp. 1–16, 2025. <https://doi.org/10.1007/s42401-025-00352-2>
- [36] S. Kang, Z. Hu, L. Liu, K. Zhang, and Z. Cao, “Object detection YOLO algorithms and their industrial applications: Overview and comparative analysis,” *Electronics*, vol. 14, no. 6, p. 1104, 2025. <https://doi.org/10.3390/electronics14061104>
- [37] Y. Pan, G. Wang, and J. Yu, “Overview of deep learning YOLO algorithm,” in *Fourth International Conference on Computer Vision, Application, and Algorithm (CVAA 2024)*, SPIE, 2025, pp. 622–630. <https://doi.org/10.1117/12.3055712>
- [38] H. Yang, B. Liang, S. Feng, J. Jiang, A. Fang, and C. Li, “Lightweight UAV Detection Method Based on IASL-YOLO,” *Drones*, vol. 9, no. 5, p. 325, 2025. <https://doi.org/10.3390/drones9050325>
- [39] J. E. Gallagher and E. J. Oughton, “Surveying You Only Look Once (YOLO) multispectral object detection advancements, applications and challenges,” *IEEE Access*, 2025. <https://doi.org/10.1109/ACCESS.2025.3526458>
- [40] S. Ruan, C. Zhan, B. Liu, Q. Wan, and K. Song, “A high precision YOLO model for surface defect detection based on PyConv and CISBA,” *Sci. Rep.*, vol. 15, no. 1, p. 15841, 2025. <https://doi.org/10.1038/s41598-025-91930-z>