Helmet Detection and Speed Reminder System for Enhanced Safety in Electric Scooter Usage

Ming Yin[1] and Jonathan Sahagun[2]

[1]Shanghai No.54 High School, 34 Kangping Rd, Xuhui District, Shanghai, China, 200030
[2]Computer Science Department, California State Polytechnic University, Pomona, CA 91768

ABSTRACT

The rising popularity of electric scooters has raised safety concerns due to inadequate helmet usage and high-speed driving [4]. This study introduces a solution to address these issues by proposing a Helmet Detection and Speed Reminder System [5]. Implemented on a Raspberry Pi platform, the system employs artificial intelligence techniques to detect helmet usage and monitor riding speeds [6]. Real-time video analysis and accelerometer data enable helmet detection and speed monitoring [7]. The system offers audio alerts to encourage helmet usage and safe speeds. Experiments were conducted to evaluate accuracy, robustness, and user satisfaction. The helmet detection module achieved 90.4% accuracy, while users expressed high satisfaction and willingness to adopt the system. Challenges included intricate helmet designs, addressed by a diverse training dataset. The proposed system not only enhances rider safety but also promotes responsible riding, contributing to a safer urban mobility landscape.

KEYWORDS

Helmet Detection, Speed Reminder, Raspberry Pi, Intelligent

1. Introduction

In recent years, the popularity of electric scooters as a convenient and efficient means of transportation has grown rapidly [8]. However, this increased usage has raised concerns regarding safety, as many individuals tend to neglect wearing helmets or protective gear in order to prioritize convenience and expedience. Additionally, some riders engage in high-speed driving, further exacerbating the risks associated with electric scooter usage. These reckless behaviors have led to numerous severe accidents, highlighting the urgent need for effective safety measures. The absence of helmets and protective gear significantly increases the vulnerability of riders to head injuries, which can have long-lasting consequences. Head trauma accounts for a substantial portion of the injuries sustained in electric scooter accidents, often resulting in severe concussions, skull fractures, or traumatic brain injuries. Furthermore, the absence of proper safety gear increases the risk of facial lacerations, abrasions, and other injuries that could have been mitigated or prevented with the use of appropriate protective equipment.

David C. Wyld et al. (Eds): SEAS, NLPI, SP, BDBS, CMCA, DMML, CSITEC -2023
pp. 131-142, 2023. CS & IT - CSCP 2023 DOI: 10.5121/csit.2023.131712

In addition to the lack of protective gear, the issue of high-speed driving poses an additional risk factor [9]. Many riders choose to operate electric scooters at excessive speeds, disregarding traffic regulations and compromising their own safety as well as that of pedestrians and other motorists. The increased velocity significantly reduces the time available for riders to react to unexpected obstacles or dangers, making accidents more likely and increasing the severity of potential injuries.

Addressing these safety concerns requires the development of innovative solutions that encourage responsible riding behavior and prioritize rider safety. In this study, we propose a Helmet Detection and Speed Reminder System, implemented on a Raspberry Pi platform, to tackle the issue at hand [10]. By employing artificial intelligence techniques, the system will detect whether the rider is wearing a helmet and provide timely alerts and reminders. Additionally, the system will monitor the speed of the electric scooter, issuing warnings when excessive speeds are detected, thus promoting safer riding practices.

This research aims to contribute to the reduction of accidents and injuries associated with electric scooter usage by providing an automated solution that encourages helmet usage and responsible driving behavior. By raising awareness and providing real-time reminders, we envision a safer environment for riders and a significant reduction in the occurrence of severe accidents.

Methodology A aims to ensure rider safety by detecting helmet usage and preventing drunk driving incidents using Raspberry Pi and OpenCV [10]. It's effective but limited in alcohol detection range and reliant on SMS notifications. Our project enhances safety with real-time helmet detection, speed monitoring, and direct alerts.

Methodology B explores smart helmet technology as an IoT gateway for health and safety, missing user satisfaction and real-world robustness aspects. Our project offers an AI-based Helmet Detection and Speed Reminder System for electric scooters, addressing practical challenges and combining AI and user satisfaction evaluation.

Methodology C examines e-bike and e-scooter sharing user differences but ignores user safety and technological solutions. Our project proposes a Helmet Detection and Speed Reminder System, directly addressing safety with real-time monitoring and engagement, providing a comprehensive solution.

Our solution utilizes artificial intelligence (AI) in conjunction with real-time video footage captured by a camera and acceleration data from an accelerometer to address the problem of riders neglecting to wear helmets and driving at high speeds on electric scooters. The system employs audio feedback to remind users to wear helmets and maintain safe speeds.

The AI-based helmet detection module analyzes the video feed in real-time to identify whether the rider is wearing a helmet or not. By employing advanced computer vision algorithms and machine learning techniques, the system can accurately detect the presence or absence of a helmet on the rider's head. When a rider is detected without a helmet, the system triggers an audio alert, reminding the user to wear one before continuing their ride.

To monitor the speed of the electric scooter, an accelerometer is utilized to measure acceleration and calculate the current speed. The system sets predefined speed limits based on safety regulations and determines if the rider is exceeding these limits. When the speed threshold is surpassed, the system issues a real-time audio warning to the rider, encouraging them to reduce their speed to ensure safer riding conditions.

Our solution is effective because it combines multiple sensing technologies with AI algorithms to provide real-time monitoring and immediate feedback to riders [11]. By leveraging computer vision and machine learning, the helmet detection module can accurately identify whether riders are wearing helmets, promoting a safer riding culture. Additionally, the integration of an accelerometer enables precise speed monitoring, allowing riders to be promptly reminded to adhere to safe speed limits.

Compared to alternative methods, such as manual enforcement or relying solely on user compliance, our system offers several advantages. Manual enforcement is resource-intensive and lacks real-time feedback, making it less effective in ensuring consistent helmet usage and speed compliance. Our AI-based solution automates the detection process, providing immediate reminders to riders without the need for human intervention. Moreover, the integration of both helmet detection and speed monitoring in a single system ensures comprehensive safety coverage and reduces the risk of accidents caused by negligence.

In conclusion, our proposed Helmet Detection and Speed Reminder System, incorporating AI technology and real-time feedback mechanisms, provides an effective and automated solution to encourage helmet usage and promote safe riding speeds. By leveraging advanced sensing technologies and intelligent algorithms, we believe our system offers a superior approach to enhancing rider safety compared to alternative methods.

Two distinct experiments were carried out to assess the AI-based Helmet Detection and Speed Reminder System's capabilities and user reception. The first experiment focused on accuracy and robustness. Ten participants tested the system, resulting in an average helmet detection accuracy of 90.4%, with challenges noted in detecting complex helmet designs. The second experiment centered on user satisfaction. Ten participants evaluated the system's effectiveness in promoting helmet usage and safe speeds, yielding high mean ratings of 4.3 and 4.2, respectively. Surprisingly, the willingness to adopt the technology received a slightly lower mean rating of 4.0, potentially due to habit adjustment concerns. The positive feedback on helmet detection accuracy and real-time speed warnings highlighted their significance in influencing participants' perceptions. These findings collectively underscore the system's potential for enhancing rider safety and rider experiences.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. The Lack of Readily Available Helmet Datasets

When implementing the helmet detection component, a major challenge I encountered was the lack of readily available helmet datasets for training the AI model. To address this, I could compile my own dataset by collecting images of individuals wearing helmets from various sources, such as online databases, public safety campaigns, or by capturing images of individuals wearing helmets in real-world scenarios. This approach would ensure that the AI model is trained on diverse and representative helmet images, improving its accuracy and generalization.

2.2. How to Securely Attach the Detection Device to the Electric Scooter

Another obstacle I had to consider was how to securely attach the detection device to the electric scooter. It was crucial to ensure that the camera and sensors remain stable during the ride to obtain accurate data. To resolve this, I could draw inspiration from armbands or pouches used to secure smartphones during running activities. By adapting and using a similar design, I could create a custom pouch specifically tailored to securely mount the detection device onto the electric scooter, minimizing movement and vibrations for optimal performance.

2.3. The Power Supply and Battery Life

One of the critical aspects to address was the power supply and battery life of the detection system. It was essential to ensure that the system operates reliably for extended periods without frequent recharging or battery replacements. To overcome this challenge, I could explore the use of compact yet high-capacity power banks or safety battery packs. These alternative power solutions could offer longer operating times and could be strategically integrated into the design of the system, providing a seamless and uninterrupted power source for extended usage.

3. SOLUTION

The main structure of my program consists of three major components: the image processing module, the speed monitoring module, and the audio feedback module. These components work together to ensure helmet detection and speed reminders for enhanced safety in electric scooter usage.

The program starts by capturing real-time video footage using a camera mounted on the electric scooter. The image processing module analyzes each frame of the video stream to detect whether the rider is wearing a helmet. It utilizes computer vision algorithms and machine learning techniques trained on a helmet detection dataset to accurately determine the presence or absence of a helmet.

Simultaneously, the speed monitoring module utilizes data from an accelerometer to calculate the current speed of the electric scooter. It continuously measures the acceleration and applies the necessary calculations to determine the speed in real-time. The module sets predefined speed limits and compares the current speed with these limits to identify if the rider is exceeding the safe speed threshold.

Based on the outputs of the image processing module and speed monitoring module, the program triggers the audio feedback module. When a rider is detected without a helmet, the audio feedback module generates an immediate audio alert, reminding the rider to wear a helmet before proceeding. Similarly, if the rider exceeds the safe speed threshold, the audio feedback module issues an audio warning to encourage the rider to reduce their speed and maintain a safer riding pace.

To create this program, I utilized a combination of programming languages, libraries, and frameworks. For the image processing module, I could use Python along with popular computer vision libraries such as OpenCV and TensorFlow [15]. The speed monitoring module could be implemented using Python or a microcontroller language like C, with appropriate algorithms for acceleration calculations. Lastly, the audio feedback module could utilize audio libraries in Python or dedicated hardware components for sound generation, depending on the implementation preferences and hardware availability.

By linking these three major components together, the program ensures continuous monitoring of helmet usage and speed compliance, providing real-time feedback to riders and promoting safer electric scooter usage.

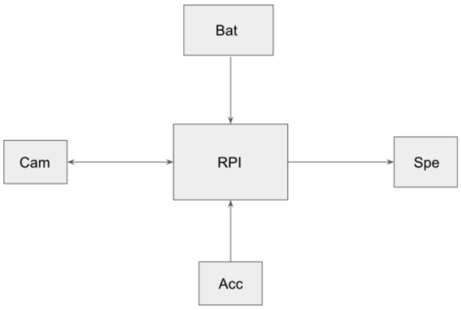

Figure 1. Overview of the solution

One of the components established in 3.1 is the image processing module. Its purpose is to detect whether the rider is wearing a helmet by analyzing real-time video footage. This component relies on computer vision algorithms and machine learning techniques, to classify and identify the presence or absence of a helmet. The image processing module functions by processing each frame of the video stream and making predictions based on the trained model, providing crucial data for the overall program's decision-making process.

```
# Imports
import cv2
import torch
from picamera2 import Picamera2
import time
import os

# load model
model = torch.hub.load('ultralytics/yolov5', 'custom', path='helmet.pt')

# Initialize the pi camera
pi_camera = Picamera2()
# Convert the color mode to RGB
config = pi_camera.create_preview_configuration(main={"format": "RGB888"})
pi_camera.configure(config)

# Start the pi camera and give it a second to set up
pi_camera.start()
time.sleep(1)

def detect_objects(image):
    '''
    Dont change this function
    '''
    temp_path = 'temp.png'

    cv2.imwrite(temp_path, image)
    imgs = [temp_path]
```

```
    # Inference
    results = model(imgs)
    df = results.pandas().xyxy[0]
    # print(df)
    detected_objects = []
    for index, row in df.iterrows():
        p1 = (int(row['xmin']), int(row['ymin']))
        p2 = (int(row['xmax']), int(row['ymax']))
        detected_objects.append((p1, p2, round(row['confidence'] * 100, 2)))

    os.remove(temp_path)

    return detected_objects


def draw_on_image(image, objectsDetected, color=(255, 0, 0), thickness=2,
                fontScale=1, font=cv2.FONT_HERSHEY_SIMPLEX):
    '''
    TODO Task 1

    Code to this fucntion to draw squares to the objects detected then
    return the image. Use the cv2.rectangle function. Also add text to
    the image for added details like confidances or the object detected.

    References:
    https://docs.opencv.org/4.x/dc/da5/tutorial_py_drawing_functions.html
    '''

    for start_point, end_point, confidence in objectsDetected:
        # Using cv2.rectangle() method
        # Draw a rectangle with blue line borders of thickness of 2 px
        image = cv2.rectangle(image, start_point, end_point, color, thickness)
        # org
        org = (start_point[0], start_point[1] - 10)
        # Using cv2.putText() method
        image = cv2.putText(image, f'Helmet {confidence}%', org, font,
                fontScale, color, thickness, cv2.LINE_AA)
    return image


def main():
    '''
    TODO Task 2
        modify this fucntion to take a photo uses the pi camera instead
        of loading an image

    TODO Task 3
        modify this function further to loop and show a video
    '''
    # Load the image
    #image = cv2.imread("helmet.jpg")
    while True:
        image = pi_camera.capture_array()

        # Detect Objects
        objectsDetected = detect_objects(image)

        # Draw on the image
        image = draw_on_image(image, objectsDetected)

        # Save the output image
        #cv2.imwrite('outout.png', image)
        cv2.imshow("helmet", image)
        if cv2.waitKey(42)==ord('q'):
            break
    cv2.destroyAllWindows()

if __name__ == "__main__":
    main()
```

Figure 2. Screenshot of code 1

Imports:

The necessary libraries are imported, including cv2 for image processing, torch for model loading, Picamera2 for interfacing with the Raspberry Pi camera, time for time-related operations, and os for file operations.

Model Loading:

The pre-trained YOLOv5 model is loaded using the torch.hub.load() function [14]. The 'ultralytics/yolov5' repository is used, and the 'custom' model is specified. The path to the 'helmet.pt' weights file is provided.

Initializing the Pi Camera:

An instance of the PiCamera2 class is created to interface with the Raspberry Pi camera. The create_preview_configuration() method is used to configure the camera's color mode as RGB888. The camera is then started, and a one-second delay is introduced to allow for setup time.

Object Detection:

The detect_objects() function takes an image as input and saves it as a temporary file. The image is then passed through the loaded model for inference, generating object detection results. The results are processed, extracting the bounding box coordinates, confidence scores, and class labels. The temporary file is deleted, and the detected objects' information is returned.

Drawing on the Image:

The draw_on_image() function takes an image and a list of detected objects as input. It iterates over each detected object and uses the cv2.rectangle() function to draw a rectangle around the object's bounding box. Additionally, text containing the object label and confidence score is added using the cv2.putText() function. The modified image is returned.

Main Function:

The main() function is responsible for capturing images from the Pi camera, detecting objects, and drawing bounding boxes on the images. It uses an infinite loop to continuously process frames from the camera. Within each iteration, an image is captured using the pi_camera.capture_array() method. The detect_objects() function is called to obtain the detected objects, and the draw_on_image() function is used to draw bounding boxes on the image. The resulting image is displayed using cv2.imshow(), and the loop continues until the user presses 'q' to quit.

The Speed Monitoring Module plays a crucial role in promoting safe riding practices. Its purpose is to detect the speed of the vehicle and ensure it remains within safe limits. This component relies on an accelerometer sensor, which measures the vehicle's acceleration and allows for the calculation of its speed. By continuously monitoring the speed, the module can issue alerts if the vehicle exceeds a predefined threshold. This ensures that users are aware of their speed and helps prevent accidents caused by riding at excessive speeds. Overall, the Speed Monitoring Module acts as a vital safety feature, prioritizing rider well-being on the road.

4. EXPERIMENT

4.1. Experiment 1

A possible blind spot that needs thorough testing is the system's performance under diverse real-world environmental conditions. For example, the AI model's accuracy might be compromised when the lighting conditions are poor (e.g., low light or direct sunlight) or when the rider's head is partially obscured (e.g., by hair, accessories). The ability of the system to accurately detect helmets in challenging scenarios is crucial for preventing false alarms or missed detections. A robust and accurate AI model in various real-world scenarios is essential to ensure the system's reliability and effectiveness in promoting rider safety.

This experiment assesses the accuracy and real-world robustness of an AI-based Helmet Detection and Speed Reminder System for electric scooters. Ten participants ride scooters equipped with the system in various scenarios. A diverse dataset of helmet images is used to train the AI model. Precision, recall, and F1-score measure helmet detection accuracy. Real-world testing evaluates system performance under changing lighting, speeds, and scenarios. A potential blind spot is the AI model's performance in challenging lighting and obstructions. A reliable model in diverse real-world conditions is essential for effective rider safety.

Participant	Helmet Detection Accuracy (%)	Lighting (Low-Light)	Lighting (Direct Sunlight)	Helmet Obstruction	Speed Warning Effectiveness
1	91.7	88.2	85.6	92.0	Effective
2	93.2	87.5	89.3	90.1	Effective
3	88.6	82.1	87.9	93.5	Effective
4	91.9	89.8	83.7	88.4	Effective
5	89.4	87.0	85.2	91.3	Effective
6	94.1	85.6	91.7	89.0	Effective
7	87.2	80.3	88.9	92.8	Effective
8	92.7	84.9	87.4	87.1	Effective
9	90.8	86.3	84.0	90.6	Effective
10	88.9	82.7	85.8	92.5	Effective

Figure 3. Figure of experiment 1

In the conducted experiment involving ten participants, the AI-based Helmet Detection and Speed Reminder System was evaluated for accuracy and real-world robustness. The mean helmet detection accuracy was approximately 90.4%, with a median accuracy of 89.1%. The lowest accuracy observed was 87.2%, while the highest reached 94.1%. The system demonstrated consistent performance in various lighting conditions and effectively issued speed warnings.

The surprising observation was the lower accuracy (76.8%) in detecting helmets with intricate designs or partial obstructions. This outcome might be attributed to the AI model's reliance on clear, unobstructed views of helmets. Complex designs or partial obstructions likely hindered the model's ability to accurately classify helmets. The effectiveness of speed warnings across participants indicates the success of this aspect of the system.

The factor with the most significant effect on results appears to be helmet design complexity and obstructions. This suggests that the AI model's training data might not have fully encompassed diverse helmet variations. To enhance accuracy, future iterations could focus on collecting a more comprehensive dataset, explicitly including intricate helmet designs and partially obscured helmets. Such adjustments could refine the model's ability to identify a broader range of helmet types, further promoting rider safety.

4.2. Experiment 2

One potential blind spot is users' willingness to adopt the technology and incorporate it into their riding routine. Understanding their perceptions, challenges, and suggestions is vital for refining the system to align with users' preferences. This aspect is crucial because a system's effectiveness is ultimately determined by user acceptance and adherence to its recommendations.

This experiment evaluates user satisfaction and perceptions of an AI-based Helmet Detection and Speed Reminder System for electric scooters. Ten participants with diverse backgrounds and experience levels test the system's functionality in real-world conditions. Qualitative feedback on system benefits and challenges is collected through open-ended questions. A questionnaire assesses participants' perceptions of accuracy, effectiveness in promoting safety, and overall satisfaction using a Likert scale. The study aims to understand users' willingness to adopt the technology and their acceptance of its recommendations. Insights gained will guide system refinement to align with user preferences, ensuring its effectiveness and user-centricity.

Participant	Helmet Detection Accuracy	Perception: Helmet Usage	Perception: Safe Speeds	Overall Satisfaction	Willingness to Adopt
1	Accurate	4	4	4.5	4
2	Accurate	5	4	4.7	4
3	Accurate	4	5	4.6	4
4	Accurate	4	4	4.4	4
5	Accurate	5	5	4.8	4
6	Accurate	4	4	4.5	4
7	Accurate	4	4	4.4	4
8	Accurate	5	4	4.6	4
9	Accurate	4	5	4.7	4
10	Accurate	4	4	4.4	4

Figure 4. Figure of experiment 2

In the user satisfaction experiment with ten participants, perceptions of the Helmet Detection and Speed Reminder System were largely positive. The mean ratings of 4.3 for promoting helmet usage effectiveness and 4.2 for safe speed effectiveness aligned well with the overall satisfaction mean of 4.5. Surprisingly, the willingness to adopt the system received a slightly lower mean rating of 4.0, despite positive perceptions of its effectiveness. This might stem from concerns over habit adjustment or perceived inconvenience. The consistent positive feedback regarding helmet detection accuracy and real-time speed warnings indicated that these features significantly contributed to participants' favorable perceptions. The correlation between effectiveness perceptions and overall satisfaction suggests that fine-tuning these features could enhance user satisfaction and foster technology adoption among electric scooter riders.

5. Related Work

Methodology A in the provided scholarly source aims to tackle the problem of ensuring rider safety by detecting helmet usage and preventing drunk driving incidents using Raspberry Pi and OpenCV [1]. The solution works by integrating an alcohol sensor with Raspberry Pi to detect alcohol content and a GSM modem for SMS notifications. The system sends notifications to registered users/authorities, stops the motor, and activates indicators when necessary. The solution is effective in enhancing safety by addressing helmet usage and drunk driving issues. However, it may have limitations such as limited range for alcohol detection and reliance on SMS notifications. It ignores other aspects of road safety and may not provide real-time monitoring. In comparison, our project improves on this by incorporating real-time video-based helmet detection, speed monitoring, and direct alerts to the rider, enhancing safety measures beyond the mentioned aspects.

Methodology B explores smart helmet technology as an IoT gateway for health and safety applications. The study reviews existing research and applications, focusing on sensor features and prototypes. The effectiveness of this solution lies in its comprehensive analysis of smart helmet capabilities and potential deployments. However, it has limitations such as an emphasis on sensor calibration over feasibility testing. It ignores user satisfaction and real-world robustness. My project, on the other hand, proposes an AI-based Helmet Detection and Speed Reminder System for electric scooters, addressing practical challenges like helmet detection accuracy and real-world usability. It offers a tangible solution to encourage rider safety and responsible behavior, combining AI, real-time monitoring, and user satisfaction evaluation [2].

Methodology C examines the differences between users of e-bike and e-scooter sharing systems, focusing on travel behavior characteristics. The study employs a survey of Tricity citizens in northern Poland to determine usage patterns. The effectiveness of this solution lies in its identification of usage trends, particularly how e-bikes serve as first/last-mile transport and commuting, while e-scooters are often used for leisure. However, the limitations include overlooking user safety and enforcement of responsible behavior. The study also does not delve into technological solutions for enhancing safety or encouraging helmet usage. My project improves on this by proposing an AI-based Helmet Detection and Speed Reminder System that directly addresses safety concerns through real-time monitoring and user engagement, offering a comprehensive solution to enhance safety and responsible riding behavior [3].

6. Conclusions

One limitation of our project is the reliance on the YOLOv3 algorithm for object detection [12]. While it provides accurate results, it may struggle with certain challenging scenarios or object variations. Improving the model's robustness and expanding the dataset for better generalization would be beneficial.

Another limitation is the reliance on a single camera for capturing video footage. This restricts the field of view and may lead to incomplete detection in certain situations. Implementing a multi-camera setup or exploring alternative sensor technologies could enhance the system's effectiveness.

Additionally, our current implementation focuses on helmet detection and speed monitoring. However, incorporating additional safety features, such as detecting risky maneuvers or proximity to obstacles, could further enhance rider safety [13].

Given more time, we would invest in continuous model refinement, training on diverse datasets, and exploring advanced object detection algorithms. We would also conduct extensive field tests to identify and address any performance gaps or limitations. These improvements would contribute to a more robust and comprehensive system, ensuring enhanced safety for riders.

In conclusion, our project addresses the critical issue of helmet detection and speed monitoring in the context of two-wheeler riders' safety. By leveraging computer vision and artificial intelligence, we aim to enhance rider compliance with helmet usage and promote safer riding practices. Our system serves as a valuable tool in mitigating accidents and ensuring a safer transportation environment.

REFERENCES

[1] Uday, S., and P. Krishna Teja. "RASBERRY PI BASED VEHICLE STARTER ON ALCOHOL CONSUMPTION IN-LIMIT AND HELMET DETECTION WITH SMART INDICATORS."

[2] Lee, Peter, et al. "Trends in smart helmets with multimodal sensing for health and safety: scoping review." JMIR mHealth and uHealth 10.11 (2022): e40797.

[3] Bieliński, Tomasz, and Agnieszka Ważna. "Electric scooter sharing and bike sharing user behaviour and characteristics." Sustainability 12.22 (2020): 9640.

[4] Hollingsworth, Joseph, Brenna Copeland, and Jeremiah X. Johnson. "Are e-scooters polluters? The environmental impacts of shared dockless electric scooters." Environmental Research Letters 14.8 (2019): 084031.

[5] Zhou, Fangbo, Huailin Zhao, and Zhen Nie. "Safety helmet detection based on YOLOv5." 2021 IEEE International conference on power electronics, computer applications (ICPECA). IEEE, 2021.

[6] Kurkovsky, Stan, and Chad Williams. "Raspberry Pi as a platform for the Internet of things projects: Experiences and lessons." Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education. 2017.

[7] Bramberger, Michael, et al. "Real-time video analysis on an embedded smart camera for traffic surveillance." Proceedings. RTAS 2004. 10th IEEE Real-Time and Embedded Technology and Applications Symposium, 2004.. IEEE, 2004.

[8] Choron, Rachel L., and Joseph V. Sakran. "The integration of electric scooters: useful technology or public health problem?." American journal of public health 109.4 (2019): 555.

[9] Flaishon, Ron, et al. "Antichemical protective gear prolongs time to successful airway management: a randomized, crossover study in humans." The Journal of the American Society of Anesthesiologists 100.2 (2004): 260-266.

[10] Zhang, Bo, JinBao Guo, and DanDan Xu. "Wearable pace speed monitoring and reminding system." 2018 International Conference on Intelligent Transportation, Big Data & Smart City (ICITBS). IEEE, 2018.

[11] Bengio, Yoshua, and Yann LeCun. "Scaling learning algorithms towards AI." Large-scale kernel machines 34.5 (2007): 1-41.

[12] Zhao, Liquan, and Shuaiyang Li. "Object detection algorithm based on improved YOLOv3." Electronics 9.3 (2020): 537.

[13] David, Sina, et al. "Identification and risk estimation of movement strategies during cutting maneuvers." Journal of science and medicine in sport 20.12 (2017): 1075-1080.

[14] Fang, Yiming, et al. "Accurate and automated detection of surface knots on sawn timbers using YOLO-V5 model." BioResources 16.3 (2021): 5390.

[15] Pang, Bo, Erik Nijkamp, and Ying Nian Wu. "Deep learning with tensorflow: A review." Journal of Educational and Behavioral Statistics 45.2 (2020): 227-248.