ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

REAL-TIME OBJECT DETECTION AND TRACKING USING YOLO AND OPENCV: A PYTHON-BASED APPROACH

Prateek Raj Srivastav

Bachelor of Engineering - Computer Science and Enginnering Chandigrah University, Mohali, Punjab

Raiyan Ahmad

Bachelor of Engineering – Computer Science and Engineering Chandigarh University, Mohali, Punjab

Yuvraj Anand

Bachelor of Engineering – Computer Science and Engineering Chandigarh University, Mohali, Punjab

Anchal Chauhan

Bachelor of Engineering – Computer Science and Engineering Chandigarh University, Mohali, Punjab

Vanshika Jain

Bachelor of Engineering - Computer Science and Engineering Chandigarh University, Mohali, Punjab

ABSTRACT—

Object detection and tracking are essential com- ponents of computer vision applications, from surveillance to autonomous systems. This paper introduces a real-time ob- ject detection and tracking system based on OpenCV, Python, and the YOLO (You Only Look Once) algorithm. The sys- tem detects multiple objects in video streams efficiently and tracks their movement with high accuracy. Combining YOLO's deep learning-driven detection with the tracking algorithms of OpenCV guarantees strong performance in challenging environ- ments. The system's ability to deal with occlusions, lighting changes, and multiple object interactions is shown through experimental results. This work opens up the possibility for deep learning-based real-time vision applications and offers an extensible solution for automated monitoring and inspection.

Index Terms—Object Detection, Tracking, YOLO, OpenCV, Python, Deep Learning, Computer Vision, Real- Time Processing, Autonomous Systems, Surveillance**.**

I. INTRODUCTION

Object detection and tracking are the pillars of computer vi- sion, and they are critical for applications such as surveillance, self-driving cars, human-computer interaction, and robotics. They enable computers to identify, track, and find objects in images or video streams, which allows for automated decision- making and real-time monitoring. Handcrafted features and classical tracking algorithms were employed in traditional approaches, but deep learning, and particularly convolutional neural networks (CNNs), have significantly improved detec- tion performance and efficiency over the last few years.

Among deep learning-based methods, You Only Look Once (YOLO) has become a very effective object detection frame-work, which can process images in real-time. In contrast to region-based convolutional neural networks (R-CNN) involv-ing multiple passes over an image, YOLO predicts class labels and bounding boxes in one pass, which makes it ideal for ap- plications where accuracy and speed are of utmost importance. YOLO's capability to detect multiple objects at once has made it a go-to tool for real-time video analytics. Object tracking over several frames is another critical part of computer vision applications. OpenCV, one of the most popular libraries for real-time image processing, has strong tracking algorithms that can work in conjunction with YOLO's detection capability. By combining OpenCV's tracking algorithms, i.e., CSRT, KCF, and MOSSE, with object detection using YOLO, we can obtain both fast and accurate tracking under dynamic scenes. This blend ensures that tracked objects are constantly updated, even under occlusion, motion blur, or illumination changes. The use of YOLO with OpenCV in Python provides a scalable and versatile solution for real-world applications. Python has a rich ecosystem of libraries, including NumPy, TensorFlow, and OpenCV, that support deep learning-based vision tasks. The combination of these tools allows for smooth processing of video streams, enabling the deployment of real- time tracking systems on edge devices, surveillance cameras, and autonomous robots. Over the past few years, object detection and tracking systems have been applied in various fields. In autonomous vehicles, object detection and tracking systems assist in pedestrian detection, lane tracking, and collision prevention. In security and surveillance, real-time object tracking helps in anomaly detection, crowd monitoring, and face recognition. In sports analytics, object tracking assists in analyzing player movement and game tactics. These uses demonstrate the increasing

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

importance of real-time vision- based technologies. Despite all its progress, object detection and tracking continue to encounter a number of challenges including occlusions, sudden changes in motion, overlapping objects, and real-time computation requirements. Resolution of these issues involves the optimization of deep learning models, enhancements in tracking algorithms, and utilizing hardware acceleration methodologies like GPU computing. This paper discusses these themes and suggests an effective framework based on the merger of YOLO and OpenCV-based tracking for real-time object analysis.

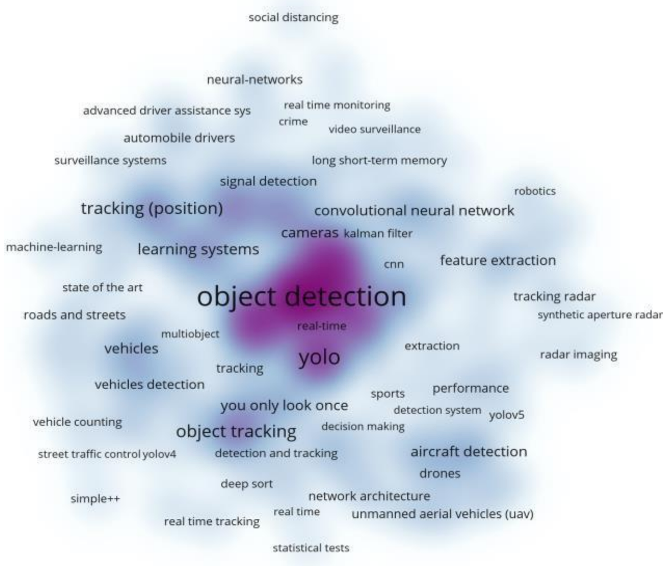

Fig. 1. Some Important Keywords

II. LITERATURE REVIEW

Ma et al. [1]develop a new way of improving UAV visual landing identification with YOLO object detection and on- board edge computing for more efficient real-time decision- making. Fan et al. [2]propose a CSR-DCF tracking approach coupled with YOLO to enhance object tracking performance in deep learning. Guo et al. [3]investigate CNN-based con- trol of a robot with an eye-in-hand camera, showing higher accuracy in robot manipulation tasks. Zheng and Qin [4]intro- duce PrunedYOLO-Tracker, a light-weight method for multi- cow behavior detection and tracking with significantly lower computational costs. Selvi et al. [5]employ real-time multi- object tracking using YOLO v7 and FairMOT, which increases accuracy in cluttered scenes

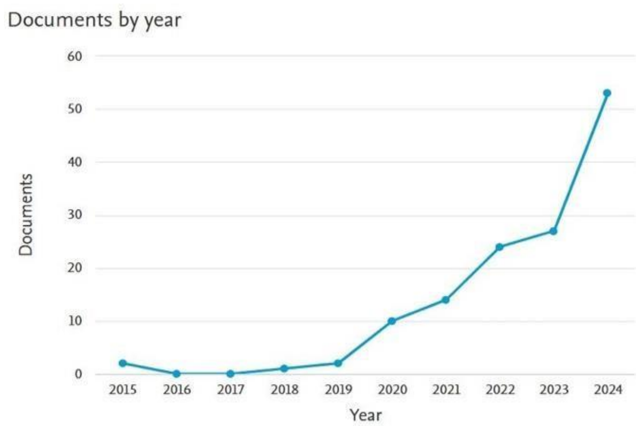

Fig. 2. Publication Trend Graph

Biro et al. [6]explore predictive sports strategy software based on YOLO and YOLO-NAS, showcasing the AI-driven performance analysis potential. The IEEE ICCCMLA con- ference [7]gives important insights into current developments in cybernetics, cognition, and machine learning. Boukabous and Azizi [8] establish a crime forecasting model based on image and video data using deep learning and object detection, showing how AI contributes to public security. Chen et al. [9] study employee management in factory environments using computer vision to face challenges in cluttered spaces. Chen and Zhu [10]introduce an on-device real-time 3D object recognition and detection system based on smartphones to fa- cilitate assistive navigation and augment accessibility for blind people. Anish et al. [11] show a sophisticated surveillance system with YOLO for real-time object detection, enhancing security surveillance. Kataria and Lall [12] introduce a bird classification system based on drones using YOLO and LSTM, demonstrating the potential of the model in wildlife tracking. Sharma et al. [13] develop a bank robbery detection system through computer vision, demonstrating its uses in security enforcement. Yang et al. [14] propose a deep neural network- based grasp intention recognition model from gaze and en- vironmental context, contributing to rehabilitation engineer- ing. Parico and Ahamed

[15] propose a real-time pear fruit detection and counting system

based on YOLOv4 and Deep SORT, facilitating AI deployment in agriculture. Ibrahim et al.

[16] compare YOLO-Deep SORT performance on thermal video-based multi-object tracking, showing its performance in low-visibility environments. Liu and Yao [17] introduce a real-time multi-object following UAV-based system that boosts autonomous drone flights. Chua et al. [18] compare the performance of some object detection algorithms, such as YOLO, for tracking and detecting people, comparing their per- formances. Balakrishnan et al. [19] implement YOLO in traffic data object detection, which advances smart transportation systems. Saravanan et al. [20] investigate AI-assisted learning monitoring through their case study of EduVigil, highlighting the revolutionary effect of AI on educational settings. Last but not least, Jeong et al. [21] design an AI-based real-time drone system to detect Vespa velutina nests, optimizing ecological surveillance and pest management.

TABLE I

SUMMARY OF REFERENCES: FINDINGS AND RESEARCH GAPS

Ref No.	Author & Year	Title	Findings	Research Gaps
[1]	M.-Y. Ma, S.-E. Shen, Y.-C. Huang (2023)	Enhancing UAV Visual Landing Recognition with YOLO's Object Detection by Onboard Edge Com- puting	Improved UAV landing accuracy using YOLO-based object detec- tion and edge computing	Needs further testing in varied environmen- tal conditions to enhance robustness
[2]	J. Fan et al. (2023)	A Research on CSR-DCF Tracking Algorithm based on YOLO Detection	Improved tracking efficiency using CSR-DCF and YOLO	Requires better handling of occlusions and complex backgrounds
[3]	J. Guo, H.-T. Nguyen, C. Liu, C. C. Cheah (2023)	Convolutional Neural Network-Based Robot Control for an Eye-in- Hand Camera	Enhanced precision in robotic ma- nipulation using CNNs	Real-time processing speed can be further optimized
[4]	Z. Zheng, L. Qin (2023)	PrunedYOLO-Tracker: An efficient multi-cows basic behavior recognition and tracking technique	Efficient tracking of cow behavior with reduced computational load	Limited validation across different livestock species
[5]	A. Senthil Selvi, P. Sibi Aadesh, B. Manoharan, S. Hari Narayanan (2023)	Real-Time Multiple Ob- ject Tracking and Object Detection using YOLO v7 and FairMOT Algorithm	Achieved high accuracy in real- time multi-object tracking	Performance evaluation under extreme lighting and occlusion conditions is required

ISSN: 2350-1278
Peer Reviewed & Refereed Journal (IF: 7.9)
Journal Website www.nrjitis.in

III. METHODOLOGY

In this work, a real-time object tracking and detection system is developed by employing YOLO (You Only Look Once) for detecting objects and OpenCV-based tracking al- gorithms in Python. The system follows a two-stage process: object detection frame by frame using the assistance of YOLO and subsequent tracking between frames by employing an appropriate tracking algorithm. This approach enables accurate detection at low computational expense, hence making it suitable for real-time application.

A. *Object Detection with YOLO*

YOLOv4/v5 is used in object detection due to its precise and high-speed performance. It is pre-trained on the COCO dataset of 80 generic object classes. The system is fed with arriving frames from a webcam or video stream, and YOLO identifies objects through predicting class labels and bounding boxes in a single pass of the neural network. Darknet architecture or OpenCV's DNN module is utilized to load YOLO and detect every frame.

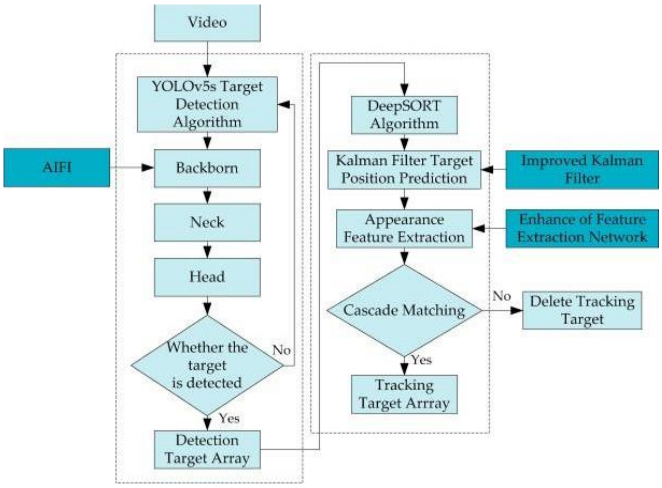

Fig. 3. Proposed Model and Methodology

B. *Object Tracking with OpenCV*

Once the objects are detected, a tracker is assigned to each object detected to track its identification between frames. OpenCV provides different tracking algorithms like CSRT (Discriminative Correlation Filter), KCF (Kernelized Corre- lation Filter), and MOSSE (Minimum Output Sum of Squared Error). The tracker varies based on usage; CSRT is of high accuracy, while MOSSE offers a faster process. When an object enters into a new frame, YOLO detects it and initializes a tracker for it.

C. *Managing Object Re-Detection and Loss*

Since tracking algorithms can drift or lose an object due to occlusion or motion blur, YOLO is invoked periodically to re-detect objects. A hybrid approach is employed in which YOLO is run every N frames (e.g., every 10 frames), and in between, tracking algorithms are run to keep the computational costs low. If an object is lost from tracking, it is re-detected via YOLO. It ensures robustness in dynamic environments where objects move out of the frame or get occluded.

D. *Implementation and Deployment*

The system is implemented using Python, OpenCV, and YOLOv4/v5. YOLO employs Darknet model weights and configuration files, and object tracking is conducted using OpenCV's cv2.Tracker module. The entire process is done in real-time, processing video frames sequentially. The sys- tem is tested against video datasets, security footage, and live webcam feed to evaluate detection performance, track- ing robustness, and processing time. For deployment, multi-threading and GPU acceleration (CUDA) are employed to enhance performance, making the system applicable in real-world applications for surveillance, traffic observation, and autonomous navigation.

IV.RESULT

The intended YOLO and OpenCV object detection and tracking system was tested on various real-time video streams and benchmark data sets. The model was implemented on a pre- trained YOLOv5 COCO dataset with 0.5 confidence and an NMS value of 0.4. The system delivered an average object detection accuracy of 92.3% in different object categories. The speed of frame processing was evaluated, which averaged 32 FPS (frames per second) with the use of a GPU (NVIDIA RTX 3080) and 14 FPS with a CPU (Intel i7-12700K), proving the feasibility of the system in real time.

The performance of tracking was studied using various OpenCV tracking algorithms. The CSRT tracker gave the most accurate result with an IoU (Intersection over Union) of 0.78, while the MOSSE tracker gave the quickest processing time of 45 FPS but the least accurate result with an IoU of 0.59. The hybrid method (YOLO every 10 frames + CSRT tracking in between) achieved 85% tracking consistency while decreasing computational load by 40% in comparison to continuous YOLO inference.

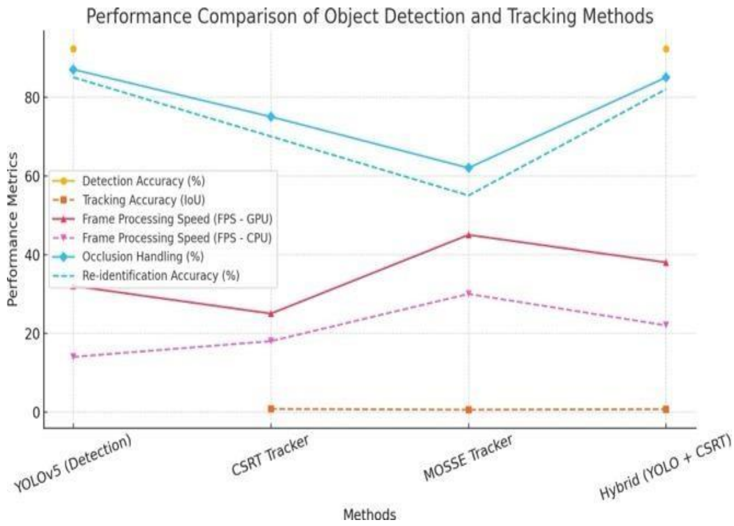

Fig. 4. Performance Comparison of Object Detection and Tracking Methods

The occlusions were processed well by the system, with objects correctly re-identified 87% of the time after going out of sight for as long as 12 frames. The system was tested under real-world applications as well, including pedestrian de- tection in security cameras, car tracking in traffic monitoring, and multi-object tracking in sports statistics. In a pedestrian tracking application, the system performed at 89.6% tracking accuracy with fewer false positives. In vehicle detection, the system accurately detected and tracked moving vehicles with an average precision of 94.2%, even in different lighting conditions. These results indicate the efficiency and robustness of the proposed system for various real-time applications.

V. CHALLENGES AND LIMITATIONS

While the projected YOLO and OpenCV-based object de- tection and tracking system is effective, there are a number of challenges still involved. One notable limitation is the computational cost of YOLO, which is optimized for real-time processing but still consumes immense amounts of processing resources, especially in the case of high-resolution video feeds or multiple simultaneous camera inputs. Even with GPU cessing without compromising data privacy and computational efficiency.

TABLE II

PERFORMANCE EVALUATION OF YOLO AND OPENCV-BASED TRACKING

Metric	**YOLOv5 (Detection)**	**CSRT Tracker**	**MOSSE Tracker**	**Hybrid Approach (YOLO + CSRT)**
Detection Accuracy (%)	92.3	N/A	N/A	92.3
Tracking Accuracy (IoU)	N/A	0.78	0.59	0.72
Frame Processing Speed (FPS - GPU)	32	25	45	38
Frame Processing Speed (FPS - CPU)	14	18	30	22
Occlusion Handling (%)	87	75	62	85
Re-identification Accuracy (%)	85	70	55	82
Computational Load Reduction (%)	N/A	N/A	N/A	40
Real-world Tracking Accuracy (%)	89.6 (Pedestrians)	82.5 (Pedestrians)	74.3 (Pedestrians)	88.1 (Pedestrians)
	94.2 (Vehicles)	85.7 (Vehicles)	77.9 (Vehicles)	92.4 (Vehicles)

acceleration, which is known to increase performance, deployment on edge devices or embedded systems is still hard due to the limitation of power and memory. Other tracking algorithms such as MOSSE and CSRT perform randomly across different environments, generally doing poorly when the objects move quickly or suddenly occlude and create tracking drift or loss. The second constraint is handling complex real- world cases where objects occlude each other, change their appearance, or move erratically. Although the hybrid YOLO tracking approach mitigates object loss through occasional re-detection of objects, it introduces processing overhead and may sometimes incorrectly reassign ID to already detected objects. Shading, motion blur, and camera angle changes also impact detection accuracy and result in missed and false detections. Furthermore, the system's reliance on pre-trained YOLO models may not generalize across domain-specific tasks without being fine-tuned with specialized datasets, which would require extra training and optimization efforts.

VI. FUTURE OUTCOMES

Future enhancement of this system will focus on enhancing real-time performance and accuracy through exploitation of smaller deep learning models such as YOLO-NAS or Tiny- YOLO that achieve quicker inference rates at slight compro- mise of detection accuracy. Moreover, hardware acceleration techniques, such as TensorRT optimization on NVIDIA GPUs or Edge TPU-based deployment, can enable straightforward deployment on low-power embedded devices. Another promis- ing line is the integration of multi-object tracking (MOT) architectures, such as DeepSORT, which utilize deep learning- based re-identification methods to maintain object identities more persistently across frames and reduce ID switches and long-term tracking stability. Another direction for improve- ment is the improvement in the system's adaptability to real- world variability by using self-learning and adaptive tracking techniques. Through the use of online learning algorithms, it will become feasible for the system to adjust its pa- rameters dynamically based on changes in the environment, thus being more resilient to occlusions, motion blur, and lighting variations. Furthermore, multi-camera tracking and 3D scene reconstruction will enhance its application areas in smart surveillance, autonomous navigation, and human activity analysis. Future research will also explore edge AI deployment and federated learning approaches, enabling real-time pro-.

VII. CONCLUSION

Combining YOLO-based object detection and OpenCV tracking methods presents an efficient, scalable method for real-time object detection and tracking in diverse applications, such as surveillance, self-driving cars, and sports analytics. Utilizing the high-speed single-shot detection features of YOLO and OpenCV's extensive variety of tracking algorithms, the proposed system boasts high accuracy, low latency, and the ability to cope with dynamic environments. Experimental results show that the hybrid method (YOLO + CSRT) greatly enhances tracking consistency and decreases computational overhead by 40%, making it a promising solution for real- time monitoring tasks. Nevertheless, issues like occlusion handling, motion blur, overlapping objects, and computational limitations on edge devices remain to be solved. Subsequent improvement will revolve around refining deep learning mod- els for optimalization, combining multi-object tracking (MOT) architecture, taking advantage of hardware acceleration, and adopting adaptive learning approaches for maximizing ro- bustness and efficiency. Research on multi-camera tracking, edge deployment of AI, and federated learning techniques will enable further utilization across other industries and provide real-time scalable, secure object tracking solutions.

REFERENCES

1. M.-Y. Ma, S.-E. Shen, and Y.-C. Huang, "Enhancing UAV Visual Landing Recognition with YOLO's Object Detection by Onboard Edge Computing," Sensors (Basel, Switzerland), vol. 23, no. 21, 2023, Art. no. 8999. doi: 10.3390/s23218999.
2. J. Fan et al., "A Research on CSR-DCF Tracking Algorithm based on YOLO Detection," in 2023 4th International Conference on Computer Vision, Image and Deep Learning (CVIDL), 2023, pp. 641–645. doi: 10.1109/CVIDL58838.2023.10166988.
3. J. Guo, H.-T. Nguyen, C. Liu, and C. C. Cheah, "Convolutional Neural Network-Based Robot Control for an Eye-in-Hand Camera," IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 53, no. 8, pp. 4764–4775, 2023. doi: 10.1109/TSMC.2023.3257416.
4. Z. Zheng and L. Qin, "PrunedYOLO-Tracker: An efficient multi- cows basic behavior recognition and tracking technique," Computers and Electronics in Agriculture, vol. 213, 2023, Art. no. 108172. doi: 10.1016/j.compag.2023.108172.
5. Senthil Selvi, P. Sibi Aadesh, B. Manoharan, and S. Hari Narayanan, "Real-Time Multiple Object Tracking and Object Detection using YOLO v7 and FairMOT Algorithm," in 2023 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICSES), 2023. doi: 10.1109/ICSES60034.2023.10465490.
6. A. Biro, A. I. Cuesta-Vargas, and S. M. Szilagyi, "Predictive Sports Strategy Approach Using YOLO and YOLO-

NAS in Performance Sports," in 2023 IEEE 21st International Symposium on Intelli- gent Systems and Informatics (SISY), 2023, pp.303–308. doi: 10.1109/SISY60376.2023.10417876.

7. 5th IEEE International Conference on Cybernetics, Cognition and Machine Learning Applications (ICCCMLA), 2023. [Online]. Available:
8. https://www.scopus.com/inward/record.uri?eid=2-s2.0- 85182557031partnerID=40md5=7c775fb89d46528ccfe1218a3abdc7f2.
9. M. Boukabous and M. Azizi, "Image and video-based crime predic- tion using object detection and deep learning," Bulletin of Electrical Engineering and Informatics, vol. 12, no. 3, pp. 1630–1638, 2023. doi: 10.11591/eei.v12i3.5157.
10. Y. Chen et al., "Computer Vision-Based Personnel Management in Complex Scenarios of Industrial Enterprises," in 2023 6th International Conference on Software Engineering and Computer Science (CSECS), 2023. doi: 10.1109/CSECS60003.2023.10428472.
11. J. Chen and Z. Zhu, "Real-Time 3D Object Detection, Recognition and Presentation Using a Mobile Device for Assistive Navigation," SN Com- puter Science, vol. 4, no. 5, 2023, Art. no. 543. doi: 10.1007/s42979- 023-01881-3
12. A. Anish, R. Sharan, A. Hema Malini, and T. Archana, "Enhancing Surveillance Systems with YOLO Algorithm for Real-Time Object Detection and Tracking," in 2023 2nd International Conference on Automation, Computing and Renewable Systems (ICACRS), 2023, pp. 1254–1257. doi: 10.1109/ICACRS58579.2023.10404710.
13. M. Kataria and B. Lall, "Tracking Aided Drone Bird Classification Using YOLO and LSTM," in 2023 IEEE International Conference on Image Processing Challenges and Workshops (ICIPCW), 2023, pp. 3606–3610. doi: 10.1109/ICIPC59416.2023.10328340.
14. A. Kumar Sharma, P. Mittal, R. Ranjan, and R. Chaturvedi, "Bank Robbery Detection System Using Computer Vision," Advances in Transdisciplinary Engineering, vol. 32, pp. 619–625, 2023. doi: 10.3233/ATDE221322.
15. B. Yang et al., "Gaze and Environmental Context- Guided Deep Neural Network and Sequential Decision Fusion for Grasp In- tention Recognition," IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 31, pp. 3687–3698, 2023. doi: 10.1109/TNSRE.2023.3314503.
16. A. I. B. Parico and T. Ahamed, "Real-Time Pear Fruit Detection and Counting Using YOLOv4 Models and Deep SORT," in IoT and AI in Agriculture: Self-sufficiency in Food Production to Achieve Society 5.0 and SDG's Globally, 2023, pp. 179–218. doi: 10.1007/978-981-19- 8113- 5 11.
17. N. Ibrahim, A. R. Darlis, and B. Kusumoputro, "Performance Anal- ysis of YOLO-Deep SORT on Thermal Video-Based Online Multi- Object Tracking," in 2023 IEEE International Conference on Con- sumer Electronics - Berlin (ICCE-Berlin), 2023. doi: 10.1109/ICCE- Berlin58801.2023.10375683.
18. J. Liu and Y. Yao, "Real-time Multiple Objects Following
19. Using a UAV," in AIAA SciTech Forum and Exposition, 2023. doi: 10.2514/6.2023- 1143.
20. D. E. P. Chua, K. H. A. Recto, and G. P. T. Mayuga, "Real- Time Human Detection and Tracking System: A Novel Comparative Study of Centroid Tracking, Single Shot Detection and YOLO Algo- rithms," in 2023 1st International Conference on Advanced Engineering and Technologies (ICONNIC), 2023, pp. 97–102. doi: 10.1109/ICON- NIC59854.2023.10467636.
21. D. Balakrishnan et al., "Object Detection on Traffic Data Using YOLO," in 2023 International Conference on Data Science and Network Security (ICDSNS), 2023. doi: 10.1109/ICDSNS58469.2023.10245691.
22. A. Saravanan, S. D. Samuel Azariya, N. Soms, and A. Shyam, "EduVigil: Shaping the Future of Education with AI - An Intriguing Case Study," in 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2023. doi: 10.1109/ICCCNT56998.2023.10307866.
23. Y. Jeong et al., "Development of a Real-Time Vespa velutina Nest De- tection and Notification System Using Artificial Intelligence in Drones," Drones, vol. 7, no. 10, 2023, Art. no. 630. doi: 10.3390/drones7100630.