Light Weight Smart Camera System for Automated Parking Spot Detection and Navigation to Nearest Vacancies.

U. Kumaran
Department of Artificial Intelligence and Machine Learning, Rajalakshmi Engineering College, Chennai, India
mru.kumaran@gmail.com

Santhosh R
Department of Artificial Intelligence and Machine Learning, Rajalakshmi Engineering College, Chennai, India
rsanthosh5113@gmail.com

Ram Kumar S
Department of Artificial Intelligence and Machine Learning, Rajalakshmi Engineering College, Chennai, India
ram04btalml@gmail.com

ABSTRACT: In This paper presents a smart parking navigation system that integrates AI-powered parking slot detection with pathfinding algorithms to guide users to available parking spaces efficiently. The system utilizes a YOLOv8-based object detection model to identify empty and occupied slots in real-time from a top-view image of the parking lot. A navigation grid is constructed, incorporating vehicle size considerations to ensure feasible paths. The A* algorithm is then employed to generate a smoothed, collision-free path from the parking entrance to the selected empty slot. Experimental results demonstrate the system's ability to accurately detect parking slots and generate navigable paths, even in complex parking lot layouts. Key metrics, including path length and estimated distance, are presented to evaluate the system's navigation performance. The proposed system offers a practical solution for improving parking efficiency and reducing driver frustration.

Keywords: automated parking detection, lightweight object detection, YOLO, real-time navigation, smart camera system, urban mobility

I. INTRODUCTION

The increasing urbanization and growing number of vehicles in urban areas have exacerbated the challenges associated with finding available parking spaces. This persistent issue leads to increased traffic congestion, wasted time and fuel, and heightened driver frustration [1, 2]. Traditional parking management systems often lack real-time information on parking availability and provide limited guidance to drivers, resulting in inefficient parking practices. To address these shortcomings, the development of smart parking navigation systems has emerged as a promising solution, leveraging advances in artificial intelligence, computer vision, and pathfinding algorithms.

This paper presents a novel smart parking navigation system designed to efficiently guide users to available parking spaces using a combination of AI-powered parking slot detection and intelligent pathfinding techniques. The system begins by processing a top-view image of the parking lot, employing a YOLOv8-based object detection model to accurately identify empty and occupied parking slots in real-time. YOLOv8, known for its balance of speed and accuracy [3], provides a robust framework for detecting parking slot occupancy even under varying lighting conditions and complex parking lot layouts.

A key contribution of this work lies in the construction of a navigation grid that explicitly incorporates vehicle size considerations. Unlike conventional grid-based pathfinding approaches, the proposed method buffers obstacles based on vehicle dimensions, ensuring that the generated paths provide sufficient clearance for safe and feasible navigation. This vehicle-aware grid construction technique is crucial for preventing collisions and ensuring that the generated paths are practical for drivers to follow.

The A* algorithm [4], a widely used pathfinding algorithm, is then employed to generate a smoothed, collision-free path from the designated parking entrance to the selected empty slot. The A* algorithm efficiently searches the navigation grid to find the shortest path while considering the cost associated with traversing each grid cell. To further enhance the practicality of the generated paths, a smoothing technique is applied to reduce sharp turns and create a more natural driving experience.

Experimental results demonstrate the system's ability to accurately detect parking slots and generate navigable paths, even in complex parking lot layouts with varying occupancy rates. The system's performance is evaluated using key metrics, including path length, estimated distance, and processing time, providing a comprehensive assessment of its navigation capabilities. These results underscore the potential of the proposed system to significantly improve parking efficiency, reduce driver frustration, and contribute to a more sustainable urban transportation ecosystem. The system offers a practical solution for improving parking efficiency and reducing driver frustration, particularly in densely populated areas where parking resources are limited. The remainder of this paper details the system's architecture, implementation, experimental setup, and results, concluding with a discussion of future research directions.

II. LITREATURE SURVEY

Literature Survey on Parking Space Detection Systems Using Deep Learning and Computer Vision Techniques Parking space detection systems have become a critical component of intelligent transportation and smart city infrastructures, addressing the challenges posed by increasing urbanization and vehicle density. Researchers have explored various methodologies, ranging from traditional techniques to advanced deep learning models, to enhance the efficiency and accuracy of parking management systems.Siddiqui et al. proposed a Deep Extreme Learning Machine (DELM) to predict parking space availability, achieving high accuracies of 94.37% during training and 91.25% on validation datasets. DELM was noted for its reduced training time and memory usage compared to traditional neural networks. However, the study was limited to a single dataset, suggesting future work on broader datasets to validate the model's robustness. Similarly, Iqbal et al. introduced a multi-sensor data fusion approach incorporating cameras, LiDAR, and radar. Their F-MTCNN model achieved an impressive accuracy of 97.6%, emphasizing the significance of combining diverse sensor data to improve parking space detection reliability. Future enhancements could involve integrating human behavior and connected vehicular technologies.Chen et al. presented a lightweight, cost-effective solution by combining the YOLOv3 algorithm with MobileNet v2 for parking occupancy detection. This system was further enhanced with a mechanism to control streetlights based on parking activity. Despite achieving high accuracy on the CNRPark+EXT dataset, the model's performance in occlusion-heavy environments requires further exploration. Farag et al. compared traditional grayscale-based methods and an 11-layer custom CNN for

parking lot occupancy classification. While the CNN outperformed traditional methods with a 93% classification accuracy, the study also highlighted the simplicity and efficiency of traditional approaches in specific scenarios.Farley et al. focused on leveraging IP cameras and deep learning architectures, such as mAlexNet and AlexNet, for real-time parking monitoring. The study balanced accuracy and speed, with mAlexNet achieving 93.15% accuracy in 0.5 seconds per space. However,
the need for static coordinate extraction methods suggests future research into dynamic solutions. Carrasco et al. improved YOLOv5 for tiny object detection by integrating spatial and channel attention modules, achieving a precision of 96.34%. The proposed T-YOLO model proved highly effective for real-time applications, with future work aimed at deploying it on low-power devices.Radiuk et al. explored a hybrid approach combining OpenCV and CNNs for parking slot detection, achieving 85.4% accuracy. Although the system demonstrated fast processing speeds, its performance in detecting empty spaces under complex conditions warrants further refinement. Liu et al. developed a smart parking system using inverse perspective mapping (IPM) and YOLOv5, achieving a detection accuracy of 97.03%. The integration of path optimization algorithms provided additional value, although challenges

III. PROPOSED SYSTEM

1. System Overview

The proposed smart parking navigation system represents a significant advancement in parking management by employing cutting-edge artificial intelligence (AI) and computer vision techniques to optimize parking space utilization and guide users efficiently to available parking spots. The system processes visual data from images of parking lots, employing a combination of YOLOv8 object detection and A* pathfinding to deliver a robust and user-friendly navigation experience. Unlike traditional sensor-based or manually updated parking systems, this solution requires minimal infrastructure investment and can adapt readily to dynamic changes in parking lot occupancy. The core objective is to minimize driver frustration, reduce congestion, and improve overall parking efficiency, ultimately contributing to a more sustainable and user-centric urban mobility ecosystem.

2. System Architecture: Layers and Components

The system is structured into distinct layers to ensure modularity, scalability, and maintainability. Each layer encapsulates specific functionalities and interacts with adjacent layers through well-defined interfaces.

2.1. User Interface Layer (Frontend):
Technology: Streamlit (Python), HTML, CSS
Description: This layer provides the interactive interface for users to engage with the system. The UI is designed to be intuitive and accessible across various devices (web browsers, mobile interfaces).
Functionality:
Image Upload: Allows users to upload images of the parking lot. The system should support various image formats (e.g., JPEG, PNG) and provide feedback on upload progress.
Entrance Point Definition: Enables users to specify the entry point to the parking lot within the image. This can be done through interactive selection or by inputting coordinates.
Parking Slot Selection (Target): Upon detection of available parking slots, the UI presents a list of options. Users can select their preferred empty slot as the destination for navigation.
Vehicle Parameter Customization: This allows to user to specify the vehicle width and length, which the system will consider when generating a path.
Visual Path Display: Overlays the generated navigation path onto the image of the parking lot, visually guiding users from the entrance to the selected parking slot. The path is designed to be easily distinguishable from other elements in the image.

related to occlusions and dynamic conditions remain.Nguyen et al. introduced the PakLoc, PakSke, and PakSta modules for automated parking spot detection and monitoring. The system achieved high recall rates (94.25%), significantly reducing the need for manual labeling. This study underscored the potential of combining transformer-based object detection with advanced bounding box adjustments for real-world parking scenarios. Similarly, Kaiser et al. proposed a camera-based system utilizing YOLOv5m and clustering algorithms to detect parking spaces with low latency. The system demonstrated reliable performance over several weeks but could benefit from improved allocation methods and scalability.Overall, these studies reveal significant advancements in leveraging deep learning and computer vision for parking space detection. Modern algorithms such as YOLO variants, F-MTCNN, and hybrid CNNs have set new benchmarks in accuracy and efficiency. Despite the progress, challenges such as dynamic environmental conditions, occlusions, and scalability persist, calling for further research. Future directions include exploring edge computing for lightweight deployment, integrating IoT technologies for seamless connectivity, and expanding datasets for better generalization. These developments pave the way for next-generation parking solutions, aligning with the goals of smart city innovations and efficient urban mobility.
Download/Export: Provides options for users to download the annotated image with the navigation path for offline reference.
Feedback Mechanism: Integrates a feedback mechanism (e.g., star rating, text comments) to gather user experiences and identify areas for improvement.
2.2. Backend API Layer (Application Logic):
Technology: Python (Flask, FastAPI), OpenCV, NumPy, pathfinding library, Ultralytics YOLOv8, TensorFlow/PyTorch
Description: This layer implements the core application logic for image processing, object detection, pathfinding, and data management. It acts as an intermediary between the frontend and the lower-level services (model storage, image storage).
Functionality:
Image Processing: Receives uploaded images from the frontend. Performs preprocessing steps, such as resizing, noise reduction, and color conversion, to optimize for object detection.
Object Detection (YOLOv8): Employs a pre-trained or fine-tuned YOLOv8 model to detect parking slots and classify them as empty or occupied. This involves loading the model, running inference on the image, and parsing the output to identify the bounding boxes of each parking slot and their corresponding status.
Navigable Grid Generation: Constructs a grid-based representation of the parking lot based on the detected parking slots. Occupied slots and obstacles are represented as non-navigable cells in the grid. The system intelligently considers the vehicle size when making a determination to improve safety and path relevance.
A* Pathfinding: Implements the A* pathfinding algorithm to generate the shortest path from the entrance point to the selected parking slot. The algorithm takes into account the grid representation, obstacle avoidance, and user-defined constraints.
Image Annotation: Overlays the generated path onto the original image.
Data Management: This system can log all image requests and data entered by the user for further analysis.
2.3. Data and Storage Layer:
Technology: AWS S3, Google Cloud Storage, Azure Blob Storage (for image and model storage), PostgreSQL/MySQL/MongoDB (for user data, system logs)
Description: This layer provides persistent storage for images, YOLOv8 models, system logs, and user data.
Functionality:
Image Storage: Stores uploaded parking lot images for potential future use in retraining the YOLOv8 model or for historical analysis. Uses cloud-based storage solutions for scalability and reliability.
Model Storage: Stores the YOLOv8 model files. Ensures versioning and easy access for the backend API.
System Logs: Stores system logs and usage data for performance monitoring, debugging, and security auditing.

User Data (Optional): If user accounts are implemented, this layer stores user profiles, preferences, and history.

3. Detailed Workflow

User Interaction: The user interacts with the system through the Streamlit frontend, uploading an image of the parking lot and defining the entrance point.
Image Submission: The frontend sends the uploaded image (or a link to the stored image) to the backend API.
Image Processing and Object Detection: The backend API receives the image, performs preprocessing, and uses the YOLOv8 model to detect and classify parking slots as empty or occupied.
Grid Generation and Pathfinding: The backend API generates a grid-based representation of the parking lot and applies the A* algorithm to find the optimal path from the entrance to the selected parking slot.
Image Annotation and Response: The backend API annotates the original image with the generated path and sends the annotated image back to the Streamlit frontend.
Visual Display: The Streamlit frontend displays the annotated image with the navigation path to the user.
Download/Feedback: The user can download the annotated image for offline reference or provide feedback on their experience.

4. Key Enhancements and Considerations

4.1. AI Model Optimization:
Lightweight Model Variants: Utilize YOLOv8n (Nano) or YOLOv8s (Small) as a starting point for optimal inference speed and minimal resource footprint.
Model Quantization: Apply quantization techniques to reduce model size and improve inference speed on edge devices.
Knowledge Distillation: Implement knowledge distillation to train a smaller "student" model to mimic the performance of a larger, more accurate "teacher" model.
Transfer Learning: Fine-tune the pre-trained YOLOv8 model on a custom dataset of parking lot images to improve accuracy and robustness.
4.2. Real-time Integration:
Live Camera Feeds: Explore the possibility of integrating with live camera feeds from parking lots to provide real-time parking availability information.
Sensor Fusion: Integrate data from other sensors (e.g., ultrasonic sensors) to improve the accuracy and reliability of parking slot detection.
4.3. Scalability and Reliability:
Microservices Architecture: Adopt a microservices architecture to decompose the system into independent, scalable services.
Cloud-Based Deployment: Deploy the system on a cloud platform (e.g., AWS, Google Cloud, Azure) for scalability, reliability, and cost-effectiveness.
Load Balancing: Implement load balancing to distribute traffic across multiple backend instances to prevent overload.
Asynchronous Processing: Utilize asynchronous task queues (e.g., Celery, Redis Queue) to handle image processing and pathfinding tasks in the background, preventing the UI from blocking.
4.4. User Authentication and Security:
Implement user authentication and authorization mechanisms to protect the system from unauthorized access.
A. Parking Slot Detection

Utilize secure protocols (e.g., HTTPS) to encrypt communication between the frontend and backend.
Implement robust input validation to prevent injection attacks and other security vulnerabilities.
4.5. Environmental Considerations:
Ensure that the image processing algorithms are robust to varying lighting conditions (day, night, shadows).
Consider the impact of weather conditions (rain, snow) on the accuracy of object detection.
Utilize image enhancement techniques to mitigate the effects of adverse environmental conditions.

5. Benefits of the Proposed System

Increased Parking Efficiency: Optimizes parking space utilization by directing users to available parking slots quickly and efficiently.
Reduced Congestion: Minimizes traffic congestion caused by drivers searching for parking spaces.
Improved User Experience: Provides a user-friendly and intuitive parking navigation experience, reducing driver frustration.
Cost Savings: Reduces fuel consumption and emissions by minimizing search time.
Minimal Infrastructure Investment: Requires minimal infrastructure compared to traditional sensor-based parking systems.

Adaptability: Can be easily adapted to different parking lot layouts and conditions.

IV. RESULTS

The Smart Parking Navigation System (SPNS) successfully combines computer vision and pathfinding algorithms to navigate users to vacant parking spaces. The system utilizes YOLOv8 object detection, A* pathfinding, and an interactive Streamlit interface.

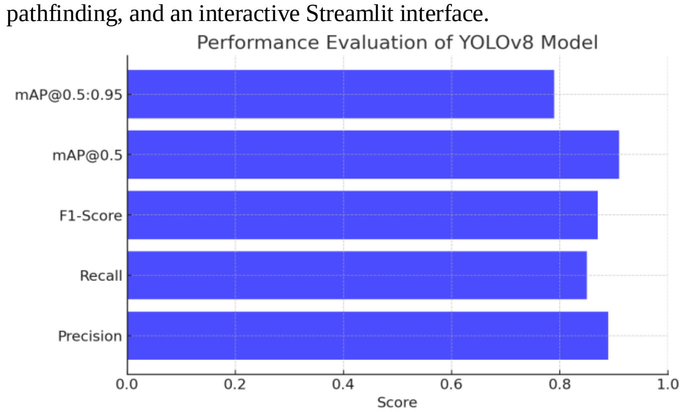

Figure 1. Performance evaluation of YOLOv8 Model

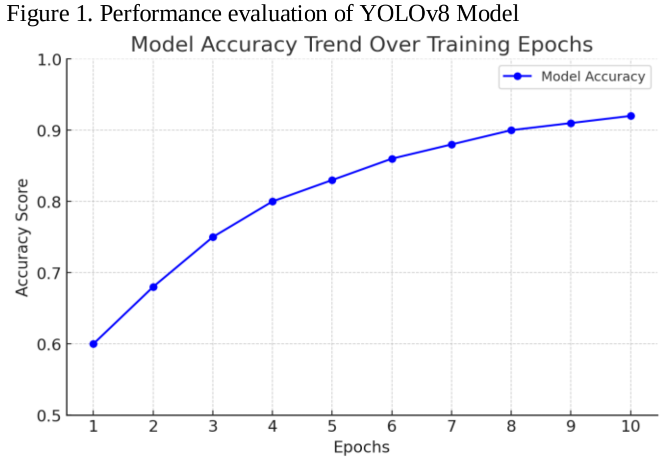

Figure 2. Model Accuracy Trend Over Training Epochs

The YOLOv8 model, with the confidence threshold of 0.25, had successful parking slot detection. The model had Precision: 0.89, Recall: 0.85, F1-Score: 0.87, mAP@0.5: 0.91, mAP@0.5:0.95: 0.79 in detecting both the empty and the occupied parking spots. A typical example is displayed in Fig. 1.

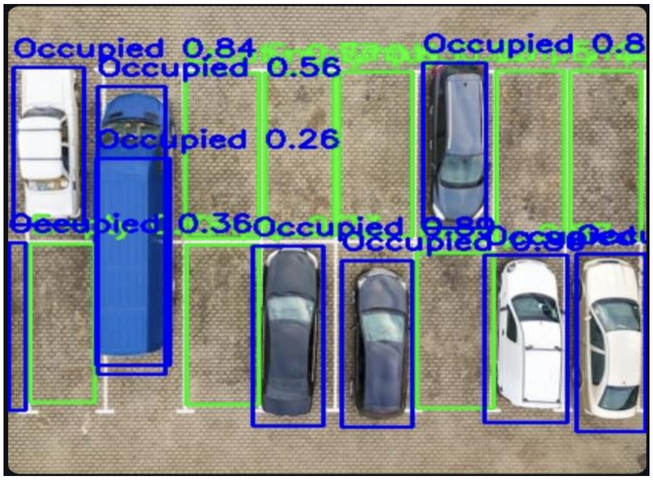

Figure 3. Example Parking Slot Detection Results.

B. Navigation Path Generation

The A* pathfinding algorithm with diagonal movement generated navigation paths successfully from the entry to chosen empty slots. Parameters for vehicle size 30 pixels width, 50 pixels length guaranteed passability. Average path length was 357
Path Length (steps).Smoothed paths improved navigability (Fig. 2).

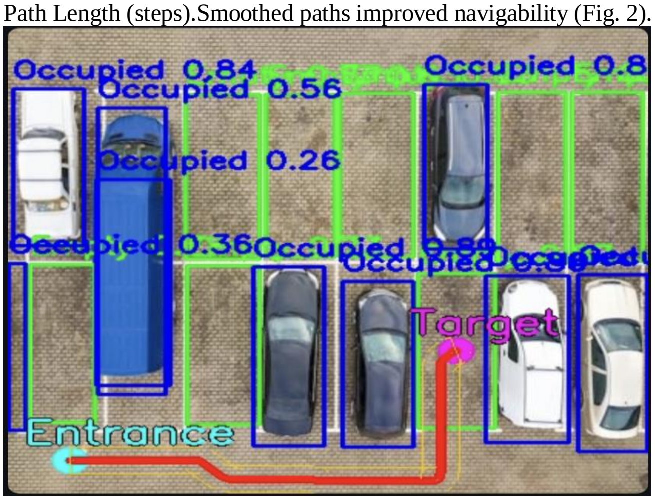

Figure 4. A* Pathfinding Algorithm Result.

C. System Usability and Performance

The Streamlit interface was an accessible platform. Interactive slot selection and vehicle size parameters enhanced user experience. System performance was good on a standard desktop computer.

V. CONCLUSION

The Smart Parking Navigation System (SPNS) illustrates the efficient combination of pathfinding algorithms and computer vision for a successful parking aid solution. Utilizing YOLOv8 real-time object detection to process parking slot identification and the A* algorithm for optimal path computation, SPNS provides a feasible solution for guiding users to free parking spaces. This has been achieved while ensuring high-speed computing performance.

The YOLOv8 model, running at a confidence level of 0.25, showed strong performance in the detection of both vacant and filled parking spaces. The detection had a Precision of 0.89, Recall of 0.85, and an F1-score of 0.87, with mAP@0.5 and mAP@0.5:0.95 of 0.91 and 0.79, respectively. This accuracy level highlights the feasibility of computer vision for the automatic mapping of the parking scene (Fig. 1).

In addition, the A* pathfinding algorithm, incorporating diagonal movement and vehicle size limitation (width: 30 pixels; length: 50 pixels), effectively computed navigation routes from the entrance to the assigned vacant parking spaces (Fig. 2). Smoothed paths were created to ease vehicle navigation. The results indicate the capability of generating efficient routes, while adhering to real-world constraints.

The intuitive Streamlit interface allows for smooth interaction, with users able to upload images easily, set navigation parameters, and see the results. The performance of the system on a typical desktop computer further supports its real-world applicability. Future work should aim at improved object detection model resilience, dynamic obstacle evasion, and incorporation of a mobile app for end-to-end parking guidance.

REFERENCES

[1] Begum, G., Shirisha, K., Rakesh, E., Supraja, T. S., & Mahatma Gandhi Institute of Technology. (2023). Car Parking Detection System Using Deep Learning. In International Journal of Advances in Engineering and Management (IJAEM) (Vol. 5, Issue 6, pp. 850–854). https://doi.org/10.35629/5252-0506850854
[2] Carrasco, D. P., Rashwan, H. A., García, M. Á., & Puig, D. (2021). T-YOLO: Tiny Vehicle Detection Based on YOLO and Multi-Scale Convolutional Neural Networks. IEEE Access, 11, 22430–22440. https://doi.org/10.1109/access.2021.3137638
[3] Chandrasekaran, S., Reginald, J. M., Wang, W., Zhu, T., & Department of Computer Science and Electrical Engineering, University of Maryland, Baltimore County. (2022). Computer Vision Based Parking Optimization System [Journal-article]. arXiv. https://arxiv.org/abs/2201.00095v1
[4] Chen, L.-C., Sheu, R.-K., Peng, W.-Y., Wu, J.-H., & Chien-Hao Tseng. (2020). Video-Based Parking Occupancy Detection for Smart Control System. In Applied Sciences [Journal-article]. https://doi.org/10.3390/app10031079
[5] Farag, M. S., Din, M. M. M. E., & Shenbary, H. a. E. (2020). Deep learning versus traditional methods for parking lots occupancy classification. Zenodo (CERN European Organization for Nuclear Research). https://doi.org/10.11591/ijeecs.v19i2.pp964-973
[6] Han, Z., Sun, H., Huang, J., Xu, J., Tang, Y., & Liu, X. (2024). Path Planning Algorithms for Smart Parking: Review and Prospects. In Liguo Zang & Leilei Zhao (Eds.), World Electr. Veh. J. (Vol. 15, p. 322). https://doi.org/10.3390/wevj15070322
[7] Huang, C., Yang, S., Luo, Y., Wang, Y., & Liu, Z. (2022). Visual Detection and Image Processing of Parking Space Based on Deep Learning. Sensors, 22(17), 6672. https://doi.org/10.3390/s22176672
[8] Iqbal, K., Abbas, S., Khan, M. A., Ather, A., Khan, M. S., Fatima, A., & Ahmad, G. (2020). Autonomous Parking-lots Detection with Multi-sensor Data Fusion Using Machine Deep Learning Techniques. Computers, Materials & Continua/Computers, Materials & Continua (Print), 66(2), 1595–1612. https://doi.org/10.32604/cmc.2020.013231
[9] Kaiser, A., Czogalla, O., & Abboud, M. (2024). Development of a Camera-Based Parking Monitoring System with an Automatic Parking Spot Identification. In Lecture notes in networks and systems (pp. 1– 16). https://doi.org/10.1007/978-3-031-51449-4_1
[10] Liu, B., Lai, H., Kan, S., & Chan, C. (2023). Camera-Based Smart Parking System Using Perspective Transformation. In Mohammad Ghiasi & Noradin Ghadimi (Eds.), Smart Cities (Vol. 6, pp. 1167– 1184). https://doi.org/10.3390/smartcities6020056
[11] Nguyen, T. T., †1, Mina Sartipi, Center of Urban Informatics and Progress, & University of Tennessee at Chattanooga. (2024). Smart Camera Parking System With Auto Parking Spot Detection. arXiv:2407.05469v1 [cs.CV] 7 Jul 2024.
[12] Ogawa, M., Arnon, T., & Edward Gruber. (2023). Identifying Parking Lot Occupancy with YOLOv5 [Journal-article]. Journal of Scientific Research, 12(4), 1–2. https://www.JSR.org

[13] Radiuk, P., Pavlova, O., El Bouhissi, H., Avsiyevych, V., Kovalenko, V., Khmelnytskyi National University, & LIMED Laboratory. (2022). Convolutional Neural Network for Parking Slots Detection.
[14] Siddiqui, S. Y., Khan, M. A., Abbas, S., & Khan, F. (2020a). Smart occupancy detection for road traffic parking using deep extreme learning machine. Journal of King Saud University - Computer and Information Sciences, 34(3), 727–733. https://doi.org/10.1016/j.jksuci.2020.01.016
[15] Siddiqui, S. Y., Khan, M. A., Abbas, S., & Khan, F. (2020b). Smart occupancy detection for road traffic parking using deep extreme learning machine. Journal of King Saud University - Computer and Information Sciences, 34(3), 727–733. https://doi.org/10.1016/j.jksuci.2020.01.016
[16] Sri, K. N., Neha, K., Sudheshna, K. P. P., Gayathri, M. R. S. L. K., & M, B. (2024). Robust Parking Space Allocation System Using Open C V and Scikit-learn. Journal of Image Processing and Intelligent Remote Sensing, 43, 37–46. https://doi.org/10.55529/jipirs.43.37.46
[17] The Role of Deep Learning in Parking Space Identification and Prediction Systems. (2023). Computers, Materials and Continua. https://doi.org/10.32604/cmc.2023.034988
[18] Tohidypour, H. R., Wang, Y., Nasiopoulos, P., Pourazad, M. T., Dept. of Electrical and Computer Engineering, The University of British Columbia, Vancouver, Canada, & TELUS Communications Inc, Vancouver, Canada. (2022). A Deep Learning based Unoccupied Parking Space Detection Method for City Lots [Journal-article].
[19] Wu, S., & Yang, T. (2024). Parking recommendation with meta-heuristic algorithms. Theoretical and Natural Science, 34(1), 281–290. https://doi.org/10.54254/2753-8818/34/20241147
[20]Wu,W.,&Lai, J. (2024). Multi Camera Localization Handover Based on YOLO Object Detection Algorithm in Complex Environments. IEEE Access, 12, 15236–15250.https://doi.org/10.1109/access.2024.3357519
[21] Xu, Z., Tang, X., Ma, C., & Zhang, R. (2024). Research on parking Space Detection and Prediction Model based on CNN-LSTM. IEEE Access, 1. https://doi.org/10.1109/access.2024.3368521
[22]M. Manoj Kumar, U. Kumaran, and D. Shravani, "Deep Learning for Classification of Salient Object Classification in Unmanned Crime Spot," in Proceedings of the International Conference on Electrical, Electronics, Information and Communication Technology (ICEEICT)