A three-stage approach using deep learning for automated vehicle smart parking with license plate recognition

Shaunak Gupta, Pushkar Garg, Abhinav Aggarwal, Gaurav Goyal and Kanu Goel*

Department of Computer Science and Engineering,
Punjab Engineering College (Deemed to be University),
Chandigarh, 160012, India
Email: shaunakgupta.bt21cse@pec.edu.in
Email: pushkargarg.bt21cse@pec.edu.in
Email: abhinavaggarwal.bt21cse@pec.edu.in
Email: gauravgoyal.bt21cse@pec.edu.in
Email: kanugoel@pec.edu.in
*Corresponding author

Abstract: Urbanisation and increasing vehicular density have amplified challenges in conventional parking systems, necessitating innovative solutions for effective parking management. Common challenges include the absence of real-time information on available parking slots, limited data analysis tools for space utilisation, reliance on manual processes, and the inadequate incorporation of automation at entry and exit of vehicles. The proposed smart parking approach endeavours to mitigate these challenges by integrating advanced technologies to enable efficient parking slot detection, comprehensive data analysis, and seamless automation. It integrates automation at entry and exit points, streamlining the parking process. Leveraging advanced computer vision and sensor technologies, the system provides real-time identification of empty parking slots, enhancing overall space utilisation. The system also incorporates accident tracking mechanisms to enhance safety within parking facilities. The research paper also presents a novel accident detection model attaining a commendable accuracy of 94.5%.

Keywords: artificial intelligence; smart parking; deep learning; automated number plate recognition; ANPR; pre-trained models; convolutional neural network; CNN.

Reference to this paper should be made as follows: Gupta, S., Garg, P., Aggarwal, A., Goyal, G. and Goel, K. (xxxx) ‘A three-stage approach using deep learning for automated vehicle smart parking with license plate recognition’, *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. x, No. x, pp.xxx–xxx.

Biographical notes: Shaunak Gupta is a BTech Computer Science and Engineering student at PEC, Chandigarh, India. His areas of research encompass machine learning, deep learning, image processing and large language models.

Pushkar Garg is a BTech Computer Science and Engineering student at PEC, Chandigarh, India. His research work focuses on machine learning, deep learning, image processing, and natural language processing.

Abhinav Aggarwal is a BTech Computer Science and Engineering student at PEC, Chandigarh, India. His areas of research include machine learning, deep learning and natural language processing.

Gaurav Goyal is a BTech Computer Science and Engineering student at PEC, Chandigarh, India. His areas of research encompass machine learning, deep learning, image processing and computer vision.

Kanu Goel is currently working as an Assistant Professor in the Department of Computer Science and Engineering, Punjab Engineering College, Chandigarh. She holds a BE (Computer Engineering), ME (Computer Science and Engineering) and PhD all from Thapar Institute of Engineering and Technology, Patiala. She is member of professional societies IEEE and ACM. Her areas of research interests include machine learning, artificial intelligence, concept drift analysis.

This paper is a revised and expanded version of a paper entitled [title] presented at [name, location and date of conference].

1 Introduction

The burgeoning challenges of urbanisation in contemporary cities necessitate innovative solutions to address the multifaceted issues surrounding urban mobility. Among these challenges, the efficient utilisation of parking spaces emerges as a critical concern, demanding a paradigm shift in traditional parking management systems. In response to this pressing need, our research introduces a groundbreaking smart parking management system that not only tackles immediate congestion issues but also lays the groundwork for a sustainable and technologically advanced parking solution (Zhang et al., 2018; de Almeida et al., 2022). By seamlessly integrating the latest technologies and adhering to thoughtful design principles, this system envisions a future where private parking areas transform into hubs of efficiency, setting a new standard for parking solutions in the face of the evolving demands of modern urban environments.

The rapid urbanisation of cities worldwide has ushered in unprecedented challenges, particularly in the realm of urban mobility. As cityscapes evolve, the strain on parking infrastructure becomes increasingly evident. This research responds to the growing need for an intelligent and adaptive parking management system that goes beyond conventional approaches. The proposed smart parking system (SPS) stands out as a revolutionary solution tailored for marked parking slots, harnessing the power of advanced image processing techniques, most notably convolutional neural networks (CNNs). At the heart of the SPS lies its ability to revolutionise parking space detection through the application of CNNs. Inspired by the visual processing mechanisms of the human brain, CNNs serve as the linchpin of the image processing capabilities, extracting intricate features from parking lot images. This technological innovation enables real-time analysis and identification of available parking spaces with unparalleled accuracy, guiding users to the nearest parking slots (Heimberger et al., 2017). By mitigating congestion and optimising the parking experience, the system addresses a crucial aspect of urban mobility, ushering in a new era of efficient parking space utilisation (Puranic et al., 2016).

The integration of automatic number plate recognition (ANPR) technology further augments the capabilities of the SPS. Working seamlessly with the image processing prowess of CNNs, ANPR automates ticketing through contactless vehicle identification. This not only streamlines the entry and exit processes but also contributes to the intelligence and efficiency of the parking management system. The synergy between CNNs and ANPR marks a pivotal advancement in parking technology, enhancing user convenience and expediting parking transactions (Adewopo et al., 2023). Acknowledging the paramount importance of safety within parking facilities, our research extends the functionality of the SPS to include real-time accident detection (Kumar et al., 2023). Achieved through a network of strategically placed sensors and cameras, this dual-faceted approach works in tandem with the image processing capabilities of CNNs. Beyond enhancing parking logistics, this feature fortifies the security and safety aspects of parking infrastructure, ensuring a comprehensive approach to smart parking management (Wong et al., 2023).

As we delve into the intricate interplay between image processing, CNNs (Radiuk et al., 2022), ANPR (Du et al., 2012), and accident detection in the subsequent sections, the aim is to illuminate the transformative potential of the SPS. By harnessing the power of cutting-edge technologies, this research endeavours to redefine urban parking, setting new benchmarks for efficiency, user convenience, and safety in metropolitan environments. The ensuing sections will delve deeper into the technical underpinnings and real-world applications of these innovative parking management techniques, offering a comprehensive understanding of their transformative impact on urban mobility (Xu and Hu, 2020).

The evolution of the SPS extends beyond conventional boundaries, embracing technological frontiers to redefine the urban mobility landscape. In the upcoming sections, we will explore in-depth the intricate details of how image processing, CNNs, ANPR, and accident detection seamlessly converge to create a holistic and intelligent parking solution (Li et al., 2020). By delving into the technical intricacies and real-world applications of these innovative techniques, this research endeavours to showcase not just the theoretical underpinnings but also the practical implications for urban environments grappling with the challenges of parking management. The journey into the future of smart parking beckons, promising a transformative shift in the way we perceive, manage, and experience parking in contemporary cities (Jung et al., 2006).

1.1 Contributions

The smart parking system, featuring ANPR, parking spot detection, and accident detection to address several challenges inherent in traditional parking systems. The unique contributions of the proposed approach are summarised as follows:

- Optimised space utilisation: Traditional parking systems often suffer from inefficient space usage due to static detection methods. Unlike most research that uses bird-eye views, our approach employs cameras at an oblique angle, offering a more practical and realistic perspective. This view captures depth and side details better, enabling more accurate detection of obstacles and parking slots. As a result, our model dynamically optimises space allocation, reduces congestion, and maximises the effective use of available parking areas.

- Enhanced user experience: By incorporating ANPR technology, our system streamlines the entry and exit processes. Users enjoy a seamless parking experience as the system automatically recognises their vehicle, eliminating the need for manual ticketing or identification. Additionally, the system includes positional error handling to ensure that alphabetical characters are in their correct positions and numerical digits are in theirs, further improving reliability. This not only saves time but also enhances the overall convenience for users.
- Accident detection for safety: The inclusion of accident detection is a crucial safety feature, and while previous models have been developed, our approach offers significant advancements. Traditional parking systems lack the capability to monitor and respond to accidents in real-time. Our model, tailored specifically for parking facilities, detects accidents with a 0.94 accuracy on the TrafficNet dataset, utilising transfer learning and domain-specific fine-tuning. This allows for immediate responses such as alerting emergency services, contributing significantly to the safety of both pedestrians and vehicle occupants.
- Real-time monitoring and management: Our smart parking system offers real-time monitoring of parking spaces and the overall facility. This feature allows for efficient management of parking resources, quick identification of issues, and proactive response to changing conditions. Traditional parking systems often rely on manual monitoring, making them less responsive to dynamic situations.
- Environmental impact reduction: Through the optimisation of parking space usage and the reduction of traffic congestion within the parking facility, our model contributes to a decrease in vehicular emissions. This aligns with societal goals for sustainability and environmental conservation, promoting a greener and healthier urban environment. In conclusion, our smart parking system goes beyond the capabilities of traditional systems by combining advanced technologies to enhance efficiency, safety, and user experience. Its unique contributions to society address the limitations of conventional parking solutions and pave the way for a more intelligent and sustainable urban mobility ecosystem.

2 Related work

Praveen and Harini (2019) discuss NB-IoT technology, providing efficient communication and mobile applications for slot booking. The GSM-based approach emphasises network data transfer, while the cloud-of-things-based approach aims to reduce user parking time. Image processing and artificial intelligence-based approaches focus on license plate detection for efficient parking. Miscellaneous approaches include real-time reservation using IoT and Wi-Fi technology.

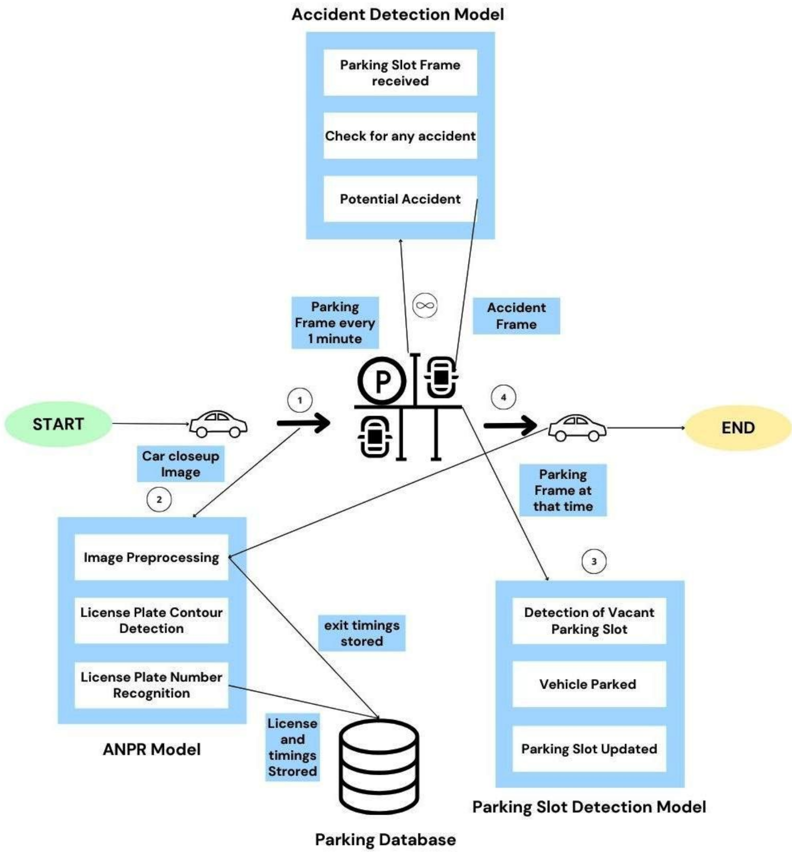
Devi et al. (2023) employ a static camera, initiating with size detection and shape analysis of car images. Unlike motion-based methods, this approach enhances efficiency in identifying images as references. Integration of deep learning minimises sensor costs and wiring complexities in the development of an intelligent parking system. Image processing techniques, including greyscale conversion, thresholding, and contour detection, are crucial for accurate comparison with a dataset, ensuring precise determination of vacant parking slots. The application of support vector machine (SVM) enhances classification accuracy, as demonstrated in the presented results. This research aligns with the trend of leveraging deep learning for effective and cost-efficient parking management systems.

A deep learning-based approach given by Kumeda et al. (2019) delves into the application of deep learning, particularly CNNs, for image classification in the context of traffic accident detection. Emphasising the power of neural networks in classification and pattern recognition through computer vision, the study focuses on the architecture of CNNs. CNNs, inspired by the human visual system, employ convolutional layers, ReLU layers, pooling layers, and fully connected layers to extract spatial features and patterns from images. The model, comprising multiple convolutional and fully connected layers, achieves notable accuracy (94.4%) on the Traffic-Net dataset, showcasing the efficacy of deep CNNs in real-world image classification tasks.

In some of the other approaches developed, smart parking systems integrate sensors and real-time data, utilising the Prewitt edge detection technique for occupancy assessment. Machine learning algorithms, particularly CNNs, enhance predictive capabilities but face challenges like data dependency (Ijjina et al., 2019; Kumeda et al., 2019). Libraries such as Numpy, CVZone, and Pickle play crucial roles in system implementation. The smart parking system employs video analysis and algorithms, utilising computer vision and Prewitt edge detection for parking space identification.

An intelligent smart parking system-based study by Alsheikhy et al. (2022) introduces a cost-effective and efficient smart parking system based on real-time image processing using a single camera. The proposed algorithm, illustrated through simulation experiments, effectively manages parking lots by detecting vacant spaces and directing drivers. The system's advantages include cost-effectiveness, environmental friendliness, easy maintenance, and integration capabilities. The algorithm employs a convolutional neural network tool for intelligent adaptation to various conditions. While fault tolerance is addressed with a backup camera, human intervention serves as an alternative.

Figure 1 Integrated block diagram of proposed approach for automated smart parking (see online version for colours)



In another work, Bouhsissin et al. (2021) compared to IoT sensor-based methods, this camera-centric approach stands out for its affordability, operational independence from internet services, and practicality.

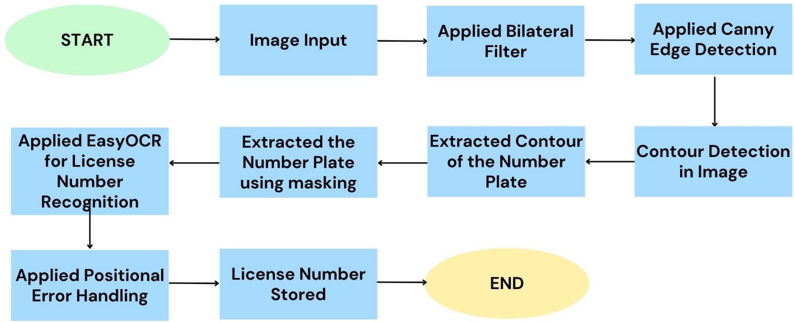
Besides the above work by Puranic et al. (2016) focuses on addressing key challenges in educational institutions related to vehicle management and theft prevention. The proposed system employs video capture and MATLAB-based application for vehicle number plate recognition using template matching. The system assumes single entry and exit gates, with initial focus on single-lane traffic. Image processing techniques, including contrast extension and median filtering, are applied for number plate localisation and character segmentation. Template matching, based on character templates, achieves an average accuracy of 80.8%. The study reviews existing methodologies, highlighting the need for a customised ANPR system tailored to educational settings. Future work includes enhancing accuracy through optimal camera positioning and incorporating neural networks for live video feed analysis. The system's potential extension to recognise multiple vehicles and implement multi-level genetic algorithms is also discussed.

3 Proposed approach

The proposed approach entails the development of an automated smart parking system through a comprehensive integration of advanced technologies. The system is designed to enhance parking management efficiency by incorporating three key phases: ANPR, parking slot detection, and accident monitoring. When a vehicle enters the parking area, the ANPR system immediately detects and identifies the vehicle's license plate. This phase is crucial for recording vehicle entry and associating it with a specific parking slot. Following this, as the vehicle occupies a parking slot, the parking spot detection feature updates the status of the slot in real-time, decrementing the count of available spaces. This dynamic update ensures an accurate and up-to-date view of parking availability. Simultaneously, real-time frames from the occupied parking slot are continuously fed into an accident detection model. This model monitors the parking space for any potential accidents or unusual events, thereby contributing to enhanced safety measures within the parking area. The integration of these phases depicted in Figure 1 which shows the automated smart parking and monitoring system.

The ANPR system's output directly influences the parking slot detection module, and both of these phases work in conjunction to feed data into the accident detection module. This tightly coupled integration allows the system to function as a cohesive unit, addressing key challenges in parking management. The result is a robust solution that provides accurate license plate recognition, real-time parking slot occupancy detection, and timely identification of incidents within the parking area.

Figure 2 Flow diagram of the proposed approach for automated number plate recognition (see online version for colours)



3.1 Recognition of license plate by ANPR system

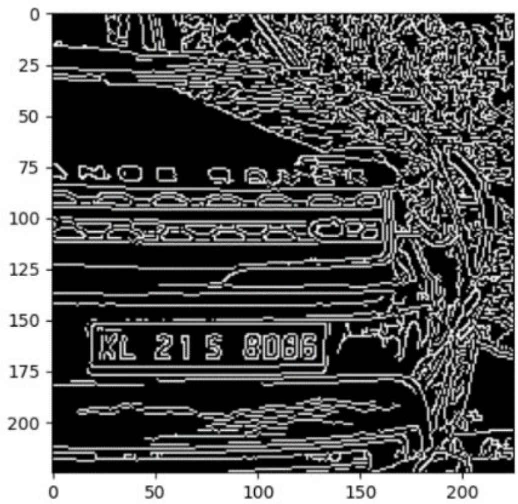
The proposed ANPR system involves a comprehensive methodology for efficient license plate recognition. The process begins with the input module, where diverse images with license plates are ingested, reflecting real-world scenarios in terms of lighting, angles, and plate types. The subsequent data collection module plays a vital role in curating a diverse dataset, ensuring the system's robustness across varied conditions. Following this, the preprocessing module applies sophisticated techniques to enhance image quality by addressing challenges like noise and variations in illumination. The flow diagram of the proposed approach for ANPR is depicted in Figure 2.

The core of the system is the number plate localisation module, utilising contour detection and approximation to identify potential license plate locations. The algorithm iterates through contours, approximating their shapes and identifying quadrilateral shapes indicative of license plates. The subsequent OCR implementation module employs optical character recognition (OCR) directly on localised license plates, converting visual alphanumeric characters into machine-readable text.

Finally, the system concludes with the output module, providing a comprehensive result that includes the recognised alphanumeric information. This output serves as a valuable resource for further processing or seamless integration into user interfaces, offering an efficient solution for ANPR.

This pseudocode involves a multi-step process for efficient license plate extraction and character recognition. Initially, contours are identified in the edge-detected image, and the top potential license plate locations are determined based on their areas. A sample license plate contour image is depicted in Figure 3. A mask is then created to isolate the license plate area, and the region of interest (ROI) is extracted from the original image. The extracted ROI is further processed using EasyOCR, a powerful OCR tool, to recognise the alphanumeric characters on the license plate. To enhance accuracy, character mapping dictionaries are employed for potential corrections. The system dynamically adjusts to variations in license plate length, accommodating both 9 and 10-character plates. This comprehensive approach ensures robustness in handling diverse real-world scenarios, such as varying lighting conditions, plate angles, and plate types, making it an effective solution for ANPR.

Figure 3 Sample license plate contour image



3.1.1 Parametric configuration

The various parametric configuration settings undertaken for different phases is stated:

1 *Bilateral filtering:*

- Diameter: Diameter of each pixel neighbourhood. The value 15 means that the filter considers pixels in a 15 × 15 neighbourhood around each pixel in the image (value: 15).
- Sigma colour: A parameter that influences the colour similarity within the pixel neighbourhood. A higher value means that more distant colours will be considered as part of the neighbourhood (value: 15).

2 *Canny edge detection:*

- Lower threshold: The lower threshold for the edges. Any gradient value below this threshold is considered not to be an edge (value: 100).
- Higher threshold: The higher threshold for the edges. Any gradient value above this threshold is

considered to be a strong edge, and values between the lower and higher thresholds are considered as weak edges (value: 300).

3 *findContours function:*

- Retrieval mode: Retrieval mode specifies how contours are retrieved. RETR_TREE retrieves all of the contours and reconstructs a full hierarchy of nested contours. This hierarchy information is useful when contours are nested or have hierarchical relationships (value: cv2.RETR_TREE).
- Contour approximation method: Specifies the contour approximation method. CHAIN_APPROX_SIMPLE compresses horizontal, vertical, and diagonal segments and leaves only their end points (value: cv2.CHAIN_APPROX_SIMPLE).

4 *OpenCV function cv2.approxPolyDP:*

- Epsilon parameter: This is the maximum distance between the original contour and its approximation. It is the precision parameter. A smaller epsilon value will result in a more accurate approximation, but it may also result in more points in the approximation (value: 10).
- Closed contour: The Boolean value specifies whether the approximated curve should be a closed contour (true) or not (false). If set to true, the function assumes the contour is closed and tries to approximate a closed curve (value: true).

3.2 Parking spot classification

The proposed approach for parking spot classification integrates a sophisticated methodology utilising YOLOv8 for car detection and spatial analysis techniques. The system starts by loading a pre-trained YOLOv8 object detection model, which processes input images and generates predictions, including bounding boxes and class indices. The subsequent steps involve image preprocessing, definition of parking areas through specified vertices, and object filtering based on class labels (specifically 'car'). Each detected car's centre coordinates are calculated, and using OpenCV's pointPolygonTest function, the system evaluates whether the car is within the predefined parking areas. The proposed architecture emphasises spatial analysis, where the positioning of detected cars relative to parking areas is systematically assessed. If a car is confirmed to be within a designated area, the system records its details and updates the count for that particular region. Visualisation techniques, such as drawing rectangles around detected cars, circles at their centres, and overlaying information about each car, contribute to a comprehensive representation of the parking scenario. The modular design allows for the adaptability of the system across diverse scenarios, enabling the definition and refinement of parking areas as needed. By leveraging YOLOv8 and spatial analysis, the proposed approach provides an effective and visually accessible solution for parking spot classification, offering a detailed understanding of car distribution within input images.

An automated smart parking system utilises computer vision techniques and the YOLOv8 object detection model. The system focuses on automating the classification and monitoring of parking spaces in a given environment. Initially, the YOLOv8 model is loaded to predict objects in input images, specifically targeting cars. The code then defines various parking areas, each represented by a set of vertices. Detected cars are filtered based on their class labels, and their spatial coordinates are analysed to determine their presence within predefined parking spaces. The system records the count of cars in each area, visually representing the results by drawing rectangles around detected cars, circles at their centres, and polygons around parking areas. The code further calculates and displays the count of occupied spaces for each area, providing valuable insights into the availability of parking spots. This comprehensive approach to parking spot classification demonstrates the potential for real-world applications, contributing to the efficiency and management of parking spaces through automated image processing and object detection techniques. Figure 4 depicts the flow diagram of the parking spot detection module.

3.2.1 Pretrained model

YOLOv8, or you only look one-level v8', is an object detection algorithm that belongs to the you only look once (YOLO) family of models. The YOLO series is known for its real-time object detection capabilities and has seen several iterations, with YOLOv8 being one of the latest as of January 2023.

Architecture of YOLOv8

- Backbone network: YOLOv8 typically employs a backbone network like CSPDarknet53, which stands for cross stage partial (CSP) Darknet 53. This is used for feature extraction from the input image.
- Neck: The network may include a neck architecture, like path aggregation network (PANet), to enhance feature representation and improve detection accuracy.
- Head: The detection head of YOLOv8 is responsible for predicting bounding boxes, object classes, and confidence scores. It uses anchor boxes to predict the bounding box coordinates.
- Output format: YOLOv8 outputs a tensor containing information about the detected objects. Each bounding box is associated with a class label, confidence score, and the coordinates of the bounding box.

Figure 4 Flow diagram of the parking spot detection module (see online version for colours)

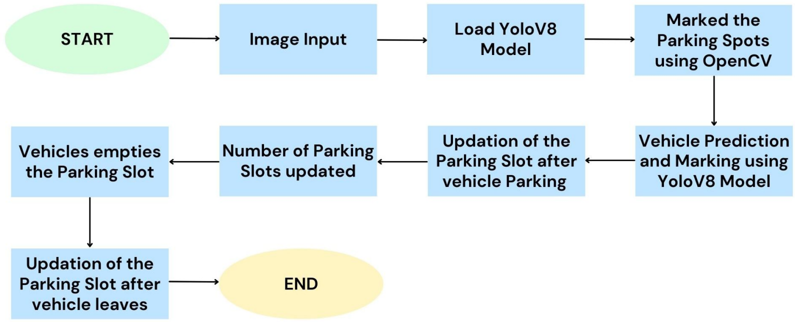


Figure 5 Proposed architecture of the convolutional neural network (CNN) for accident detection

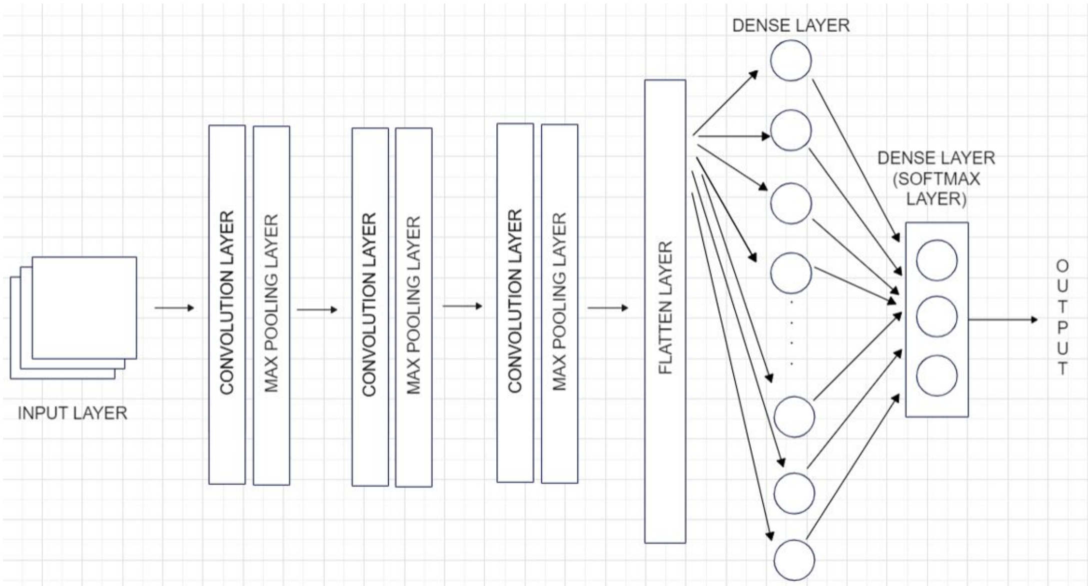


Table 1 Architecture of MobileNet base model CNN

Type/stride	*Filter shape*	*Input size*
Conv/s2	3 × 3 × 3 × 32	224 × 224 × 3
Conv dw/s1	3 × 3 × 32 dw	112 × 112 × 32
Conv/s1	1 × 1 × 32 × 64	112 × 112 × 32
Conv dw/s2	3 × 3 × 64 dw	112 × 112 × 64
Conv/s1	1 × 1 × 64 × 128	56 × 56 × 64
Conv dw/s1	3 × 3 × 128 dw	56 × 56 × 128
Conv/s1	1 × 1 × 128 X	56 × 56 × 128
Conv dw/s2	3 × 3 × 128 dw	56 × 56 × 128
Conv/s1	1 × 1 × 128 × 256	28 × 28 × 128
Conv dw/s1	3 × 3 × 256 dw	28 × 28 × 256
Conv/s1	1 × 1 × 256 × 256	28 × 28 × 256
Conv dw/s2	3 × 3 × 256 dw	28 × 28 × 256
Conv/s1	1 × 1 × 256 × 512	14 × 14 × 256
5× (Conv dw/s1 and Conv/s1)	3 × 3 × 512 dw and 1 × 1 × 512 × 512	14 × 14 × 512 and 14 × 14 × 512
Conv dw/s2	3 × 3 × 512 dw	14 × 14 × 512
Conv/s1	1 × 1 × 512 × 1,024	7 × 7 × 512
Conv dw/s2	3 × 3 × 1,024 dw	7 × 7 × 1,024
Conv/s1	1 × 1 × 1,024 × 1,024	7 × 7 × 1,024
Avg Pool/s1	Pool 7 × 7	7 × 7 × 1,024
FC/s1	1,024 × 1,000	1 × 1 × 1,024
Softmax s1	Classifier	1 × 1 × 1,000

Mathematical concepts

- Bounding box prediction: YOLOv8 uses anchor boxes to predict bounding box coordinates. The coordinates are usually represented as (x, y, w, h), where (x, y) is the centre of the box, and (w, h) are the width and height.
- Object confidence: YOLOv8 predicts the confidence that an object is present in a bounding box. This confidence score is a measure of how likely it is that the detected object is of the predicted class.
- Class prediction: The model predicts the probability distribution over all the classes for each bounding box. The class with the highest probability is assigned to the object detected in that box.
- Loss function: YOLOv8 employs a combination of localisation loss, confidence loss, and classification loss. The total loss is calculated as the sum of these individual losses. The loss function guides the model during training to improve its predictions.
- Anchor boxes: YOLOv8 utilises anchor boxes to improve bounding box prediction. These anchor boxes are pre-defined boxes with specific widths and heights. The model adjusts these anchors during training based on the dataset.

3.3 Accident detection

The proposed architecture of the CNN is designed for a classification task with four output classes as depicted in Figure 5. It consists of following components:

1 *Input layer:* The model begins with an input layer that expects images with dimensions (224, 224, 3), where 224 represents the height and width, and 3 corresponds to the RGB colour channels. This layer serves as the entry point for the input data.

2 *MobileNet base model:* The core of the architecture is the MobileNet base model, a well-known and efficient convolutional neural network architecture. This pre-trained model is included without the fully connected (dense) layers at the top. The purpose of this base model is to capture hierarchical features from the input images through a series of convolutional and depthwise separable convolutional layers. The architecture of MobileNet base model is represented in Table 1.

3 *Global average pooling 2D:* Following the MobileNet base, a global average pooling 2D layer is applied. This layer reduces the spatial dimensions of the tensor by computing the average value of each feature map. It serves as a spatial summarisation technique, consolidating the learned features across the spatial dimensions.

4 *Dropout layer:* To prevent overfitting and enhance generalisation, a dropout layer with a dropout rate of 0.5 is introduced. This layer randomly sets a fraction of input units to zero during training, thereby introducing a form of regularisation and reducing the model's reliance on specific neurons.

5 *Dense layer with softmax activation:* The final layer is a dense layer ('dense') with 4 units, indicating the number of output classes in the classification task. It is equipped with a softmax activation function, converting the raw model outputs into probabilities. The softmax function ensures that the sum of the class probabilities is equal to 1, enabling the model to make predictions across the specified classes.

In summary, this architecture leverages the features learned by the MobileNet base model, applies global average pooling for spatial aggregation, introduces dropout for regularisation, and concludes with a dense layer for classification. The model is compiled using the Adam optimiser with a learning rate of 0.0001 and employs categorical crossentropy as the loss function, while accuracy is monitored as the evaluation metric during training.

3.3.1 Pretrained model setup

MobileNet is a lightweight convolutional neural network (CNN) architecture specifically designed for mobile and embedded vision applications, where computational resources may be limited. Developed by Google researchers (Howard et al., 2017), MobileNet introduces depth wise separable convolutions to achieve a good balance between model accuracy and computational efficiency. The basic architecture of MobileNet comprises two key components: depthwise separable convolutions and linear bottlenecks.

- *Depthwise separable convolutions:* MobileNet replaces the standard convolutional layers with depthwise separable convolutions, which consist of two distinct operations: depthwise convolutions and pointwise convolutions. Depthwise convolutions apply a single filter to each input channel independently, reducing computational complexity by performing convolutions on each channel separately. Pointwise convolutions then apply 1x1 convolutions to combine the outputs from depthwise convolutions, allowing the network to learn cross-channel correlations efficiently.
- *Linear bottlenecks:* To further optimise the model, MobileNet introduces linear bottlenecks, where a 1×1 convolutional layer is followed by a linear activation function. This linear activation function helps prevent information loss during the bottlenecks and allows for better representation learning. The combination of depthwise separable convolutions and linear bottlenecks significantly reduces the number of parameters and computations, making MobileNet well-suited for resource-constrained environments. MobileNet is flexible and can be easily customised

for various applications by adjusting hyperparameters such as depth and width multipliers.

4 Dataset description

4.1 Accident detection

The dataset employed for training the accident detection module is Traffic-Net, curated by DeepQuest AI. Traffic-Net is instrumental in training machine learning systems to recognise and interpret various traffic conditions, with a specific emphasis on accident detection within the traffic environment. The version of Traffic-Net used for our research consists of 4,400 high-resolution images, categorised into four distinct classes: accident, dense traffic, fire, and sparse traffic, with each class comprising 1,100 images. To enhance the performance of the model, various augmentation techniques were applied, such as rotation, flipping, scaling, and brightness adjustments. These augmentations help to increase the robustness of the model by providing diverse training examples under different conditions, which is particularly important for accurate accident detection.

4.2 Automatic number plate recognition and parking slot detection

For the ANPR module, explicit training was not required, as it relies on image processing techniques. The Indian Vehicle License Plate Dataset from Kaggle, consisting of over 1600 labelled images, was utilised for testing purposes. These images were carefully selected to represent a variety of license plate configurations, ensuring the robustness and adaptability of the implemented algorithm. In contrast, the parking slot detection module did not require any dataset for testing, as it operates purely on image processing techniques. The decision to use the ANPR dataset aimed to evaluate the module's generalisation capabilities under diverse real-world conditions.

5 Results and discussion

5.1 Automatic number plate recognition

Upon getting the coloured picture of the front view of the car containing its number plate, it is first converted to greyscale image to simplify algorithms and reduce computational complexity as greyscale images contain only intensity information. Then, the image is passed through bilateral filtering and canny edge detection to find the biggest rectangle in the image, that is the license plate itself. Bilateral filtering is applied to reduce noise and the license plate is recognised by finding contours in the edge detected image and the final image is extracted using a mask. Finally, OCR is applied on the license plate using EasyOCR to extract the license number. This process ensures a robust recognition pipeline that efficiently handles various real-world scenarios. The ANPR module demonstrated reliable performance when tested with the Indian Vehicle License Plate Dataset from Kaggle, consisting of over 1,600 labelled images, achieving an overall accuracy of 93.4%. Various stages of ANPR for a sample image are depicted in Figure 6.

5.2 Parking slot detection

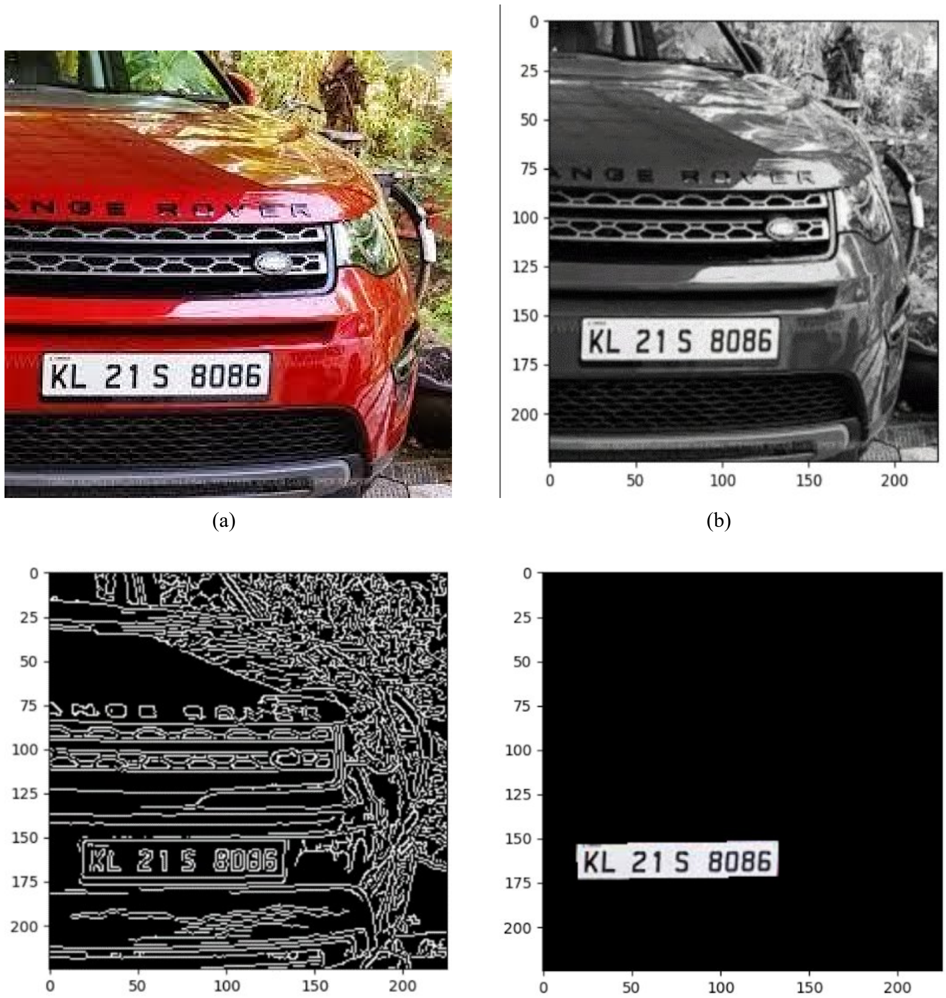
Various stages of the parking slot detection module for a sample image are demonstrated in Figure 7. Firstly, the parking spots are identified and marked using OpenCV. Figure 7(a) illustrates a clear parking area before any markings are applied. Then, the parking slots are delineated, as shown in Figure 7(b), where the identified parking spots are highlighted within the parking area. In real-time, when cars arrive, a pre-trained YOLOv8 model (yolov8s.pt) is employed to detect vehicles. The model iterates through rows of the DataFrame corresponding to the identified cars and checks whether the centroid of each detected vehicle falls within any of the pre-marked parking spots using OpenCV's pointPolygonTest function. If a car is detected within a marked spot, the respective parking spot's colour changes to indicate the presence of a vehicle, as depicted in Figures 7(c) and 7(d). Additionally, a variable named space is maintained to track the number of parking spots occupied and available out of the total number. The graph depicted in Figure 8 plots the training and validation loss over epochs for accident detection module. Two lines are plotted: one for training loss (labelled as Train) and another for validation loss (labelled as Val). The x-axis is labelled as epochs and ranges from 0 to approximately 17.5. The y-axis is labelled as loss and ranges from 0 to 0.8. Both the training and validation loss decrease sharply initially and then level off, indicating that the model is learning but then reaches a point where improvements are marginal.

The Train line is in blue, while the Val line is in orange. The graph shows the model accuracy over epochs for training and validation datasets. Two lines represent Train and Val (validation) datasets. The x-axis, labelled as epoch, ranges from 0 to approximately 17.5. The y-axis, labelled as accuracy, ranges from 0.75 to 1.00. The Train line is blue and shows a rapid increase in accuracy before plateauing just below 1.00 accuracy. The orange Val line also increases but begins to plateau at around 0.90 accuracy, showing some variance compared to the training data.

5.3 Accident detection

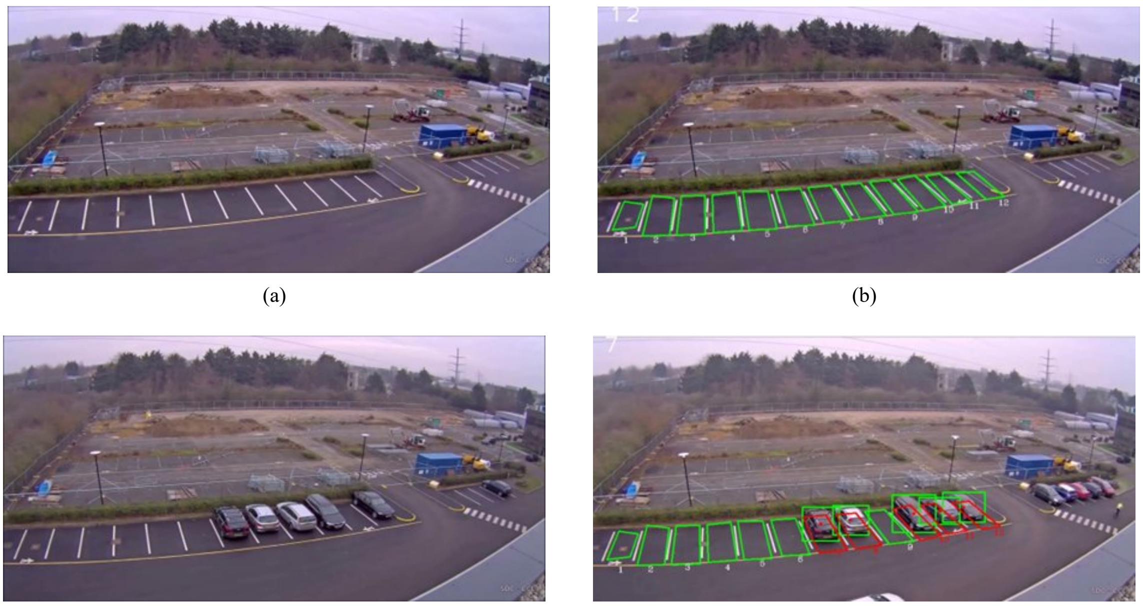
Table 2 represents the results of the evaluation matrices achieved of the proposed approach. The four rows correspond to different classes (0, 1, 2, 3) with their respective scores in each metric. Below the class rows, there are additional rows for overall model performance metrics including accuracy, macro average, and weighted average. The accuracy of the model is 0.94. Macro average and weighted average precision, recall, and f1-score are also provided with values in the range 0.94–0.95.

Figure 6 Stages for automatic number plate recognition for sample image, (a) original image (b) grey-scaled image (c) after filtering and applying Canny edge detection (d) number plate extracted from the car (see online version for colours)



(a) (b) (c) (d)

Figure 7 Stages of parking slot detection for sample image at different time slots, (a) at time t = t0 before (b) at time t = t0 after (c) at time t = t1 before (d) at time t = t1 after (see online version for colours)



(a) (b) (c) (d)

Figure 8 Accuracy plot for accident detection (see online version for colours)

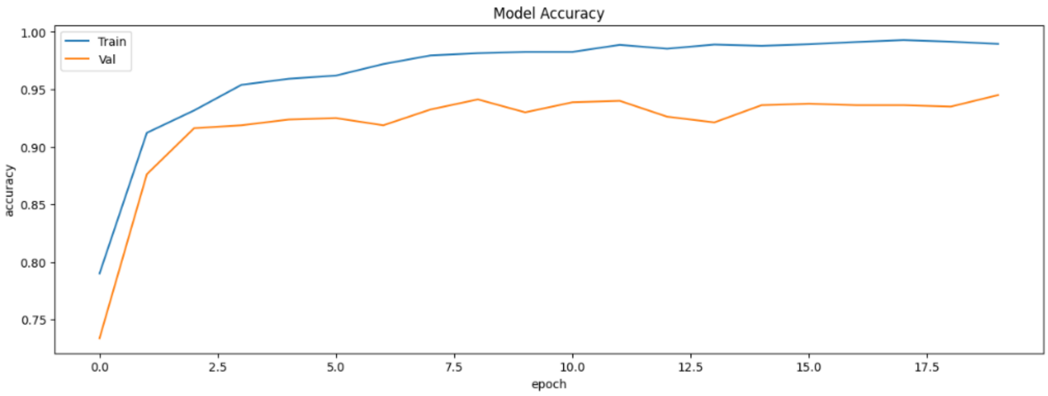


Table 2 Evaluation metrics of the proposed approach

	Precision	*Recall*	*F1-score*	*Support*
0	0.97	0.90	0.93	200
1	0.95	0.97	0.96	200
2	0.95	0.98	0.97	200
3	0.92	0.93	0.92	200
Accuracy			0.94	800
Macro avg	0.95	0.95	0.94	800
Weighted average	0.95	0.94	0.94	800

6 Conclusions and future work

In this paper, we have used features such as ANPR, parking slot detection, and accident monitoring in our SPS. These features provide a huge upgrade on the traditional parking system. ANPR is useful to the organisation for maintaining their parking record. Parking slot detection is useful to the user to find a parking spot without any manual input. Accident monitoring is useful to both the user and the organisation to find any occurrence of mild accidents within the parking space. This dual focus on convenience and safety positions the smart parking system as a comprehensive solution that not only optimises parking space utilisation but also prioritises the well-being and security of users and their vehicles. Overall, the significance of this research lies in its potential to improve current parking systems, highlighting the importance of continued exploration in this domain.

Looking into the future, there is a drive towards refining ANPR capabilities to include the recognition of license plates captured at oblique angles, ensuring adaptability in unconventional parking scenarios. Additionally, the expansion of ANPR to encompass two-wheelers, recognising the diverse array of vehicles in urban landscapes. Also incorporating various types of number plates and the different types of format of vehicle number. There is a promising avenue for the enhancement of smart parking systems to include the identification of minor incidents such as scratches and dents within parking areas among parked vehicles which is not currently possible due to lack of data. It would offer a more comprehensive level of security and awareness for vehicle owners. There is room to refine the existing YOLO model for parking slot detection by optimising its parameters and fine-tuning on a diverse dataset. Improve annotation precision and implement post-processing techniques to enhance detection accuracy. Optimise real-time performance for dynamic parking scenarios and establish a user feedback loop for continuous model improvement.

References

Adewopo, V.A., Elsayed, N., ElSayed, Z., Ozer, M., Abdelgawad, A. and Bayoumi, M. (2023) 'A review on action recognition for accident detection in smart city transportation systems', *Journal of Electrical Systems and Information Technology*, Vol. 10, No. 1, p.57.

Alsheikhy, A.A., Shawly, T., Said, Y.F. and Lahza, H. (2022) 'An intelligent smart parking system using convolutional neural network', *Journal of Sensors*, Vol. 2022, pp.1–11.

Bouhsissin, S., Sael, N. and Benabbou, F. (2021) 'Enhanced VGG19 model for accident detection and classification from video', *2021 International Conference on Digital Age & Technological Advances for Sustainable Development (ICDATA)*, IEEE, pp.39–46.

de Almeida, P.R.L., Alves, J.H., Parpinelli, R.S. and Barddal, J.P. (2022) 'A systematic review on computer vision-based parking lot management applied on public datasets', *Expert Systems with Applications*, Vol. 198, p.116731.

Devi, T.R., Hemlathadhevi, A. et al. (2023) 'A enhanced novel methodology for checking driver eligibility and vehicle security using image processing', *International Journal of Intelligent Systems and Applications in Engineering*, Vol. 11, No. 2, pp.1032–1044.

Du, S., Ibrahim, M., Shehata, M. and Badawy, W. (2012) 'Automatic license plate recognition (ALPR): a state-of-the-art review', *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 23, No. 2, pp.311–325.

Heimberger, M., Horgan, J., Hughes, C., McDonald, J. and Yogamani, S. (2017) 'Computer vision in automated parking systems: design, implementation and challenges', *Image and Vision Computing*, Vol. 68, pp.88–101.

Howard, A.G., Zhu, M., Chen, B., Kalenichenko, D., Wang, W., Weyand, T., Andreetto, M. and Adam, H. (2017) *MobileNets: Efficient Convolutional Neural Networks for Mobile Vision Applications*, arXiv preprint arXiv:1704.04861.

Ijjina, E.P., Chand, D., Gupta, S. and Goutham, K. (2019) 'Computer vision-based accident detection in traffic surveillance', *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, IEEE, pp.1–6.

Jung, H.G., Kim, D.S., Yoon, P.J. and Kim, J. (2006) 'Parking slot markings recognition for automatic parking assist system', *2006 IEEE Intelligent Vehicles Symposium*, IEEE, pp.106–113.

Kumar, K., Singh, V., Raja, L. and Bhagirath, S.N. (2023) 'A review of parking slot types and their detection techniques for smart cities', *Smart Cities*, Vol. 6, No. 5, pp.2639–2660.

Kumeda, B., Fengli, Z., Oluwasanmi, A., Owusu, F., Assefa, M. and Amenu, T. (2019) 'Vehicle accident and traffic classification using deep convolutional neural networks', *2019 16th International Computer Conference on Wavelet Active Media Technology and Information Processing*, IEEE, pp.323–328.

Li, W., Cao, H., Liao, J., Xia, J., Cao, L. and Knoll, A. (2020) 'Parking slot detection on around-view images using DCNN', *Frontiers in Neurorobotics*, Vol. 14, p.46.

Praveen, M. and Harini, V. (2019) 'NB-IoT based smart car parking system', *2019 International Conference on Smart Structures and Systems (ICSSS)*, IEEE, pp.1–5.

Puranic, A., Deepak, K. and Umadevi, V. (2016) 'Vehicle number plate recognition system: a literature review and implementation using template matching', *International Journal of Computer Applications*, Vol. 134, No. 1, pp.12–16.

Radiuk, P., Pavlova, O., El Bouhissi, H., Avsiyevych, V. and Kovalenko, V. (2022) *Convolutional Neural Network for Parking Slots Detection*.

Wong, G.S., Goh, K.O.M., Tee, C. and Sabri, A.Q.M. (2023) 'Review of vision-based deep learning parking slot detection on surround view images', *Sensors*, Vol. 23, No. 15, p.6869.

Xu, C. and Hu, X. (2020) 'Real time detection algorithm of parking slot based on deep learning and fisheye image', *Journal of Physics: Conference Series*, Vol. 1518, p.012037, IOP Publishing.

Zhang, L., Huang, J., Li, X. and Xiong, L. (2018) 'Vision-based parking-slot detection: a DCNN-based approach and a large-scale benchmark dataset', *IEEE Transactions on Image Processing*, Vol. 27, No. 11, pp.5350–5364.