Beyond Sight: VQA for Car Parking Detection Using YOLOv8

Sphooti Kulkarni[a], Anushree Bashetti[b], Samarth Nasabi[c] and Kaushik Mallibhat[d]
School of Electronics and Communication, KLE Technological University, Hubballi, India

Keywords: VQA, YOLO, Object Detection.

Abstract: Visual Question Answering (VQA), is an interesting application area of Artificial Intelligence that can enable the machines to understand the image content and answer questions about the image. VQA integrates vision-based techniques with the Natural Language Processing techniques. The VQA model uses the visual elements of the image and information from the question to generate the best possible answer. The paper demonstrates the use of state of art, object detection algorithm -You Only Look Once (YOLO) for the identification of free and occupied car slots in a car parking system. Existing VQA systems for parking often struggle with some limitations including real-time application in dynamic and varying lighting conditions, night or bad weather, and cannot handle user queries related to parking availability, thus impacting their overall usability and effectiveness in practical applications. In this paper, a car parking VQA model has been designed where both image and question of the user feed as an input to the proposed system. The image is captured by a camera installed in the parking lot on a real time basis, and users select a question from the provided question menu. The system provides a user-friendly, menu-based question-answering system that allows users to select questions of interest and receive relevant responses based on the detected parking slot information. The proposed approach utilizes a YOLOv8 model, trained on the annotated PKLot dataset to detect and count both parked and vacant slots in real-time. This detection is integrated with a menu-based question-answer system, allowing users to interact with the model and receive accurate slot information based on their selected queries. The model performs well in both day and night, even in low- light conditions, due to the diverse PKLot dataset used for training, which covers various days and weather conditions. To improve efficiency and accuracy, some images are converted to grayscale, and a custom dataset is created. This preprocessing optimizes performance for nighttime monochromatic images, enhancing results under varying lighting. Trained on the PKLot dataset, the model achieved a mean Average Precision (mAP) of 0.994 for vacant and 0.993 for parked slots. The robust performance stems from the diversity of PKLot images and strategic preprocessing. In simpler terms, To enhance the efficiency of parking, a novel approach has been adopted, which combines computer vision with questionnaires presented in a user-friendly menu style, inspired by natural language processing (NLP). This technique enables the model to accurately detect if parking spaces are available or occupied, ultimately making the parking experience more convenient and hassle-free.

1 INTRODUCTION

The term ”Visual Question Answering” refers to the process of observing an image or visual input and providing an appropriate response to a question related to that visual content. It is important to consider how a computer perceives an image or video through computer vision, and then relates the question to the image in order to provide an answer using natural language processing. VQA has gained popularity in the fields of natural language processing and computer vision because humans have the ability to view objects in images and comprehend their characteristics and behaviors. VQA is a multi-discipline research topic that has grown in popularity in natural language processing and computer vision because humans often see objects in images and understand how they interact with their qualities and states when they look at them (Lu, Ding, et al. 2023). VQA has various practical applications, ranging from medical to fashion. It tackles the complex task of combining image analysis with question understanding to generate accurate responses. In this case, it's used to determine the status

[a] https://orcid.org/0009-0006-5170-2880
[b] https://orcid.org/0009-0009-2342-2306
[c] https://orcid.org/0009-0003-0179-6493
[d] https://orcid.org/0000-0002-3610-0332

Kulkarni, S., Bashetti, A., Nasabi, S. and Mallibhat, K.
Beyond Sight: VQA for Car Parking Detection Using YOLOv8.
DOI: 10.5220/0013604100004664

In *Proceedings of the 3rd International Conference on Futuristic Technology (INCOFT 2025) - Volume 2*, pages 834-843
ISBN: 978-989-758-763-4

of parking lots. As the number of cars grows, effective parking management becomes crucial for places like universities, malls, and airports. The aim is to develop a model that accurately detects vacant and occupied parking spaces. In (Hela, Arakkal, et al. 2022), the slot detection exhibited inaccuracies and errors when operating under low- light conditions. The model is also trained to identify parking spaces at night by analyzing black-and-white images (gray scale images). Surprisingly, the research unveiled a striking similarity between nighttime CCTV images and their corresponding color images after being converted to black and white. As a result, the author can reliably detect and tally occupied and unoccupied parking spaces both in daylight conditions, utilizing RGB images captured by the camera, and at night, using grayscale images.

The novelty of this work lies in assigning unique numbers to each parking slot, allowing precise tracking of individ- ual slot occupancy. Additionally, the model is designed to perform slot detection during both day and night, using RGB and grayscale images, ensuring continuous monitoring. This capability addresses a significant challenge in parking lot management by providing reliable performance throughout the day and applicability of the system in real-world parking lot management scenarios, providing continuous and accurate detection in varying lighting conditions.

1.1 Motivation

Parking-related issues have recently gained significant attention among the general public. In urban cities, the need to address these problems has become crucial. The rising population and vehicle numbers have led to a shortage of parking spaces, creating significant challenges in finding suitable parking. This issue is further worsened by the ineffective management of existing parking facilities. This paper addresses parking as a management issue rather than just a scarcity of spaces. The lack of information on parking availability leads to resource loss and management challenges. Additionally, detecting empty parking spaces at night is crucial. To address this issue, researchers are working on a solution that involves analyzing grayscale images of parking spots captured by cameras. This data is then used as input for their model. While ground sensors are commonly used to differentiate between parking spaces, they require continuous maintenance and installation, which can be expensive.

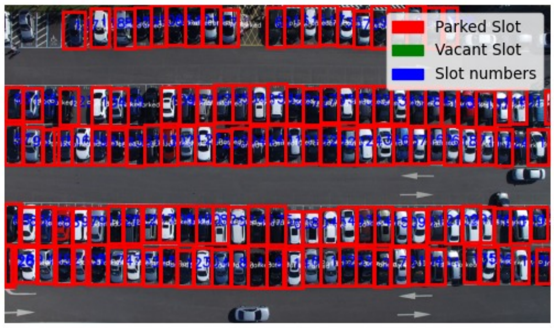

Figure 1: Car parking lot status determined using YOLOv8

2 LITERATURE SURVEY

Several literature surveys have explored the use of machine learning and deep learning models in various scenarios. In more extensive complexes where self-assisted parking is not feasible and human assistance is insufficient, alterna- tive methods are necessary. Some approaches involve ultrasonic, geomagnetic, infrared, ground sensors, vision-based systems that surpass traditional sensors, and other conventional methods (Ding, and Yang, 2019). The study of the outdoor parking lot parking space identification method is now one of the significant research projects utilizing deep learning to handle parking space detection (Li , 2022). In one study, researchers utilized the Markov Random Field framework and support vector machines to determine slot availability. Instead of employing image segmentation, they opted for different machine learning algorithms. In (Zhang, Zhao, et al. 2023), Parking spot detection is done using YOLOv5s on the PKLot dataset which achieved 99.6% accuracy. Also the custom dataset was created using the data augmentation operations such as changing bright- ness, contrast, standardization etc. by collecting images from nine different surveillance video clips and achieved accuracy of 80.38%. The drawbacks of the paper is that the model was only able to detect occupied slots and not vacant slots. In (Sairam, et al. 2020), both the occupied and vacant slots in parking lots are detected using YOLO and Mask R-CNN (Region-based Convolutional Neural Network) algorithm and achieved an accuracy of 94%. The improvements suggested to this paper were that instead of using bounding boxes, masks can be used for IoU (Intersection over Union) calculations as they are more precise and easy way to examine the degree of overlap/intersection between predicted object and ground truth object. In (Singh, and Christoforou, 2021), detections were made using a public dataset with images from surveillance cameras in various conditions. Models like MobileNet and ResNet50 were used for classification. The focus was on improving the ability to detect parking spaces without relying on visible parking lines. The authors in the paper, used VQA-based car parking occupancy detection using YOLOv5 and intent classification using RASA NLU. The model has also been tested in low-light conditions, but the predictions were incorrect. In (Huang, He, et al. 2021), the model is specifically trained to detune the car at night using Yolov3 and the MobileNet v2 network. In (Amato, Fabio, et al. 2016), the authors employed the encoder-decoder structure of SID deep learning approaches to fuse the YOLO model, which was trained with night- time data, with the SSD (Single-Slot Detector) model in order to identify employing smart camera networks. In (Duy-Linh, Xuan-Thuy, et al. 2023), it enhances the backbone, neck, and head modules of the network to improve YOLOv5 for parking lot identification in smart parking systems. These changes aim to simplify processes, improve workflow, and detection performance.

Recent advancements in automatic car parking systems highlight the versatility of YOLOv8 across diverse applica- tions. In one study, an "Advanced Car Parking System" integrated with Arduino Uno and IoT demonstrated real-time parking availability, enhancing user convenience while reducing traffic congestion (Sharmila, Rohinth, et al. 2024). A comparative survey found YOLOv8 to be the most efficient for object detection in smart car parking systems, particularly when combined with OpenCV and EasyOCR for improved image enhancement and number plate identification (Naik, Borkar, et al. 2024). Additionally, research employing YOLOv5 and CNN for Automated Number Plate Recognition (ANPR) has improved efficiency in park- ing management (Surve, Shirsat, et al. 2023). Other studies have developed smart parking systems using YOLOv5 and ResNet50, achieving high mean Average Precision (mAP) for parking space detection on low-computation devices (Balusamy, Shanmugam, et al. 2024). Notably, a novel approach utilizing YOLOv8 for real-time identification of vacant and occupied parking slots demonstrated signifi- cant improvements in parking space management (Shankar, Singh, et al. 2024). Moreover, the integration of YOLOv8 with Optical Character Recognition (OCR) technologies has shown substantial advancements in character recognition accuracy for number plate detection (Sarhan, Rahem, et al. 2024). Collectively, these findings suggest that YOLOv8 not only excels in parking-related tasks but also showcases its adaptability and efficiency across various domains.

Object detection techniques are categorized into traditional methods and deep learning-based approaches. Tradi- tional methods, such as Viola-Jones, SIFT, and HOG, are slower due to computational limitations. In contrast, deep learning mimics human brain analysis, recognizing complex patterns in images and text for accurate predictions, and automating tasks like image description and audio transcription.

1. Introduced in 2001 by Paul Viola and Michael Jones, the Viola–Jones object detection framework is a machine learning system primarily designed for face detection but adaptable to various objects. While it may not match the accuracy of convolutional neural networks, its efficiency and compact size make it suitable for scenarios with limited computational resources (Viola and Jones, 2001). Another approach in computer vision is the HOG, which counts gradient orientations in specific image areas, though it faces challenges like slow processing speed and limitations with scale and light variations in human detection (Dalal and Triggs, 2005). Additionally, David Lowe's SIFT algorithm, dating back to 1999, is widely used for identifying, characterizing, and matching local features in images. Its applications span object recognition, image stitching, 3D modeling, gesture recognition, video tracking, wildlife identification, robotic mapping, and navigation, ensuring reliable object identification through feature descriptions extracted from training images (Paolo, Andrea Prati, et al 2012).

2. R-CNN (Regions with CNN Features) revolutionized object detection by combining CNNs with region pro- posals, significantly improving accuracy (Gandhi, 2018). The SSD excels in balancing speed and efficiency for real-time object recognition, making a notable impact on applications requiring quick and precise identification. YOLO is a real-time object recognition technique known for efficiently locating and identifying multiple objects within an image or video frame, streamlining the process for enhanced speed and effectiveness (Ding and Yang, 2019).

2.1 Objectives

The project focuses on the following objectives. Propose and implement a deep learning algorithm for the detec- tion of vacant and occupied slots in a car parking system. The developed solution should be able to detect the slots both for daylight and low-light scenes. Further employ a menu-based question-answering system, facilitated by a questionnaire, to understand the user requirements and answer accordingly.

2.2 Datasets

Several available datasets include CNRPark+EXT, CARPK, NDISPark, Parking Space Detection and Classification, and PKLot available on different websites. Out of these datasets, the CNRPark+Ext dataset contains car parking lot images captured by various cameras with different perspectives and under different weather conditions. Also, the Car Parking Lot Dataset (CARPK) consists of approximately 90,000 parking lot car images collected by means of drones in order to have clearly captured images. NDISPark dataset includes 250 images of several parking areas of different situations in real-time captured using different cameras under various weather conditions and different angles. The Parking Space Detection and Classification dataset comprised images of parking lot spaces along with corresponding bounding boxes and annotations. For the model training, PKLot dataset is selected, which includes 12,416 images sourced from surveillance camera frames. These images capture different parking lots under varied weather conditions, including sunny, cloudy, and rainy days. The dataset categorizes images into either occupied or unoccupied, which aids in training the model to differentiate between these two classes effectively. To enhance the diversity of the training data, a subset of the PKLot dataset is strategically employed. Out of the 9656 images selected for training, 1242 were transformed to grayscale. This transformation emulates night-time captures from CCTV cameras, further enriching the dataset with varied lighting conditions. The remaining images retain their RGB format, contributing to a well- balanced and representative training dataset. The dataset consisted of 6767 training images, 1927 validation images, and 962 images for testing purpose.

2.3 Proposed Pipeline

The first step in the approach is to perform object detection on images capturing car parking lots from the collected PKLot dataset. This phase employs YOLOv8, a state-of-the-art object detection model. The YOLOv8 model is used for object detection due to its efficiency and high accuracy in detecting objects in real-time. The table I presents a comparison of different YOLO models. YOLOv8 is particularly well-suited for parking slot detection, as it can rapidly process images while maintaining high detection accuracy. For the object detection process, two distinct classes have been defined: ”parked” to indicate occupied slots and ”vacant” for unoccupied slots. The YOLOv8 model requires images and corresponding

Figure 2: RGB image from PKLot dataset and converted grayscale image

annotation files in the YOLO format, detailing class labels and bounding box coordinates (x, y, width, height). This YOLO algorithm systematically divides the input image into a grid of cells, generating precise bounding boxes for detected objects. The bounding boxes are used to localize objects in the image. Each predicted bounding box is associated with a confidence score, coordinates, and class predictions (vacant or parked). YOLOv8’s performance is evaluated using the mAP metric, with results showing an mAP of 0.993 for vacant slots and 0.988 for parked slots, indicating high accuracy in detecting parking slots under varied conditions. After the successful execution of object detection, the methodology transitions to the object counting phase. This step involves counting the detected objects categorized as parked and vacant slots separately. The counting process is crucial for generating answers to user queries and provides key insights into the current status of the parking lot.

With both object detection and counting achieved, the subsequent step involves integrating these results with a menu-based Question Answering system. To facilitate user interaction, a user-friendly menu featuring a set of ques- tions has been introduced. The users can select a question of interest, prompting the system to provide a relevant response. The range of questions includes various aspects related to parking, such as the count of parked and vacant slots, and the overall status of the parking lot. This system is designed to enhance the informativeness and accessibility of the parking lot management system, making it easier for users to quickly retrieve essential information

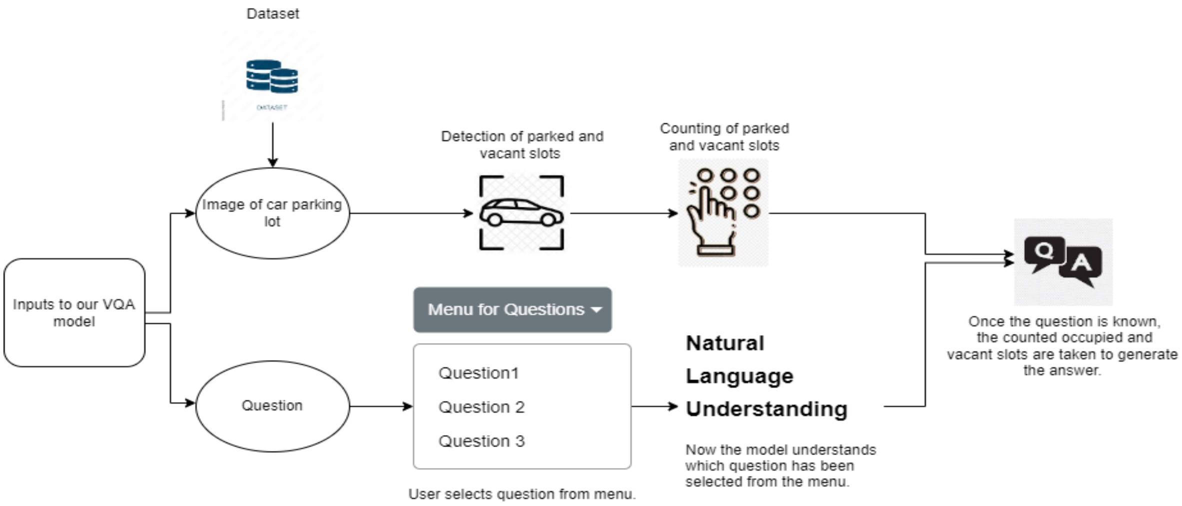

Figure 3: Proposed pipeline

about park- ing availability. The overall methodology integrates object detection, counting, and question-answering seamlessly. Once the YOLOv8 model detects and classifies parking slots, the results are passed to the question answering system, which maps the detected classes (vacant and parked) to the appropriate answers. For example, if the user asks, "How many vacant slots are there?", the system counts the instances of vacant slots (class 0) and provides the result. The Questions are displayed in Table II. To make this interaction seamless, the system is designed with a user-friendly menu, enhancing accessibility and making the solution more intuitive. This integration of object detection, counting, and interactive querying provides an effective and user-centric solution that facilitates real-time parking lot analysis and decision-making.

3 IMPLEMENTATION AND RESULTS

In order to train the model, the required dataset must contain the images of parking lots depicting clearly the parked and vacant slots. The downloaded dataset named PKLot contains a separate set of images for training, validation and testing. The annotation file (.txt file) contains [class, x center, y center, width, height] which is in YOLO format and the data is normalized. The dataset images and annotation files are uploaded in ROBOFLOW in order to get annotated dataset where image is annotated with its corresponding annotations .txt file resulting in formation of bounding boxes along with classes shown in

Table 1: Comparison Table

Method	**Accuracy**	**Ref**
YOLOv5	90%	13
CCN	100%	-
YOLOv5	94.31%	2
Custom-YOLOv5M	mAP@0.5: 97%	14
YOLOv8	94.27%	16
YOLOv8	98.70%	15
YOLOv8 (Our model)	**mAP@0.5:99.3%**	-

Table 2: Menu of questions provided to the user

Sl.No.	**Questions**
1	How many cars are parked?
2	How many vacant slots are available?
3	What is the status of the parking lot?
4	Is this slot parked or vacant?
5	Can I park my car?
6	Can I know which all slots are occupied?
7	Can I know which all slots are vacant?

Fig. 4. (class0 – vacant & class1 – parked)

The image shown in Fig. 4 is a sample image depicting how an image looks after it gets annotated in the ROBOFLOW platform, where there are 37 parked slots and 3 vacant slots. Similar annotations are done for all 9656 input images. Now a code snippet is generated which will be compiled in google colab for further training of the model. The annotated dataset is

Figure 4: Sample annotated image of PKLot Dataset from Roboflow platform

executed in google colab environment using YOLOv8 with a code snippet and the model is now trained with 5 epochs and imagsz=700 which took around 0.707 hours resulting in a) GPU mem b) box loss c) cls loss d) dfl loss e) instances and f) size as shown in Fig. 5

Fig. 5 provides an overview of the model's performance metrics after 5 epochs of training using YOLOv8 on the parking slot dataset. The graphics processing unit (GPU) memory usage during training was 11.5 GB, indicating the computational resources required for processing. The box loss (0.734) represents the error in predicting the bound- ing boxes for the parked and vacant slots, while the class loss (0.4071) captures the error in classifying the parking spaces as either vacant or occupied. Additionally, the distribution focal loss (DFL) (0.8912) corresponds to the model's confidence in assigning appropriate bounding boxes during object detection. These losses indicate the training per- formance, with lower values corresponding to better predictions. The model handled a total of 318 instances across 1922 images, with a resolution size of 704x1024 for the input images. The model's precision and recall values for detecting both vacant (class 0) and parked (class 1) slots were remarkably high, with a precision (P) of 0.99 and recall (R) of 0.993 for parked slots, and a precision of 0.993 and recall of 0.99 for vacant slots. The mAP50 (mean Average Precision at 50% IoU threshold) for both classes was nearly perfect, reaching 0.993 and 0.994, respectively. For more stringent evaluation, the mAP50-95 metric, which considers IoU thresholds between 50% and 95%, showed a slightly lower value of 0.88, which is still an excellent result, indicating the model's robustness in detecting parking slots under varied conditions and overlapping bounding boxes.

The line graphs shown in Fig. 6 depict the results of different parameters accessed for object detection technique resulting in how the model is getting trained in batch wise along with the annotated set of images. The train box loss, train class loss, and train DFL loss all show a steady decline as the number of epochs increases, indicating that the model improves its accuracy in predicting bounding boxes and classifying parked or vacant slots over time. Similarly, the validation box loss, validation class loss, and validation DFL loss follow a similar downward trend, suggesting that the model generalizes well to unseen data. The performance metrics, such as precision and recall, also show an upward trend, stabilizing as the model progresses through the epochs, reflecting improvements in the model's ability to correctly detect and classify parking slots. The mAP50 and mAP50-95 metrics demonstrate consistent improvement, indicating that the model is becoming more proficient in handling varying degrees of overlap between predicted and actual bounding boxes. These results demonstrate the efficacy of the YOLOv8 model in parking slot detection, providing high accuracy for both parked and vacant slot classification, while also maintaining strong performance across a range of challenging IoU thresholds. The decreasing loss metrics and increasing precision and recall highlight the robustness of the model in detecting parking slots in both training and validation phases.

To detect parking slots and count the number of parked and vacant classes, simple YOLO command is used i.e. results[0].boxes.cls which gives classes (0 and 1 in our case) for each detected object using which the total parked and vacant slots can be counted. After this part, there is a fixed set of questions which the user is going to ask, the menu- based question answer approach makes the user select the question as per the requirement and our model processes the image captured at that point of time and generates the output. In order to make it user friendly, GUI (Graphical User Interface) has been designed and implemented using Visual Studio Code software as shown in Fig. 8.

3.1 Measuring Outputs

The performance of the car parking slot detection model is evaluated using a variety of metrics to ensure a compre- hensive understanding of its capabilities. Key performance metrics such as precision, recall, and F1 score are utilized to assess the model's performance, and their respective curves (see Fig. 9) provide valuable insights into how well the model identifies parking slots at different confidence levels. The Precision-Confidence Curve plots the model's precision against its confidence score. Precision is the ratio of true positive detections to the total number of detec- tions (both true positives and false positives). As seen in the Fig. 9, the precision remains high across most confidence levels, indicating that the model consistently detects parking slots with high accuracy. A similar trend is observed in the Recall-Confidence Curve, which plots the recall against con-

```
      Epoch    GPU_mem   box_loss   cls_loss   dfl_loss  Instances       Size
        5/5      11.5G      0.734     0.4071     0.8912        318        704: 100% 423/423 [07:34<00:00,  1.07s/it]
                 Class     Images  Instances      Box(P          R      mAP50  mAP50-95): 100% 61/61 [01:17<00:00,  1.27s/it]
                   all       1922     112796      0.992      0.992      0.993       0.88
                     P       1922      52907       0.99      0.993      0.993      0.872
                     V       1922      59889      0.993       0.99      0.994      0.888

  5 epochs completed in 0.707 hours.
```

Figure 5: Model training with 5 epochs

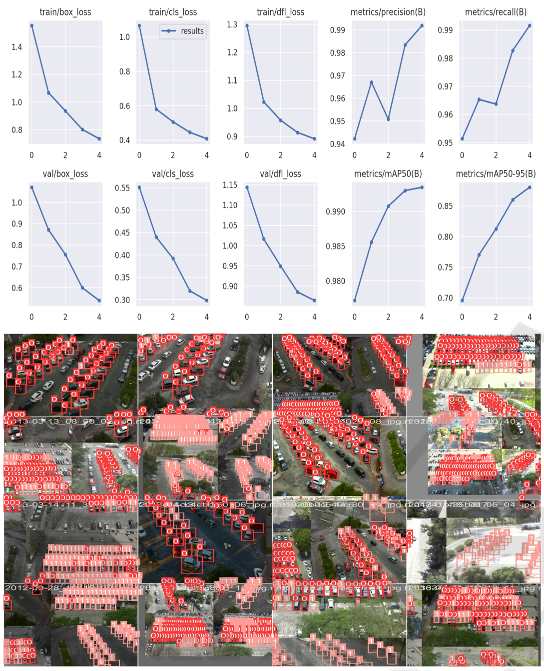

Figure 6: Results during training phase

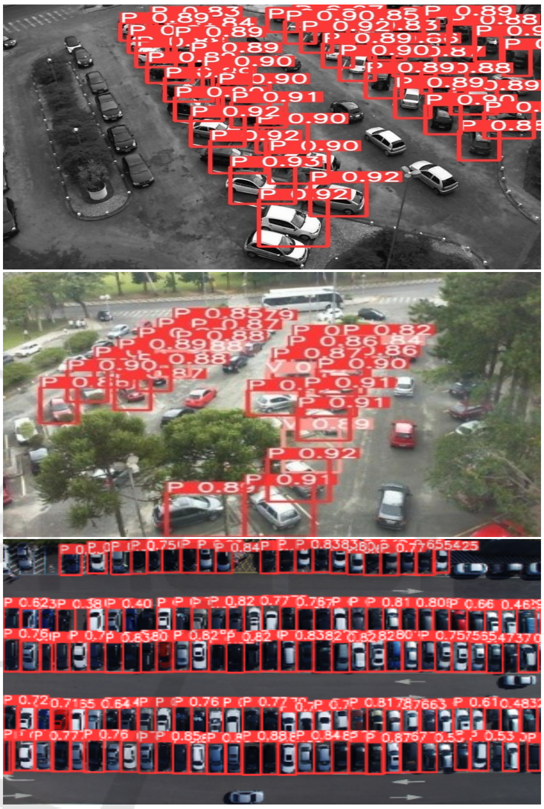

Figure 7: Results after training

fidence scores. Recall measures the model's ability to correctly identify all positive instances (in this case, both vacant and parked slots) by considering the ratio of true positives to the total number of actual positive instances. In this case, the recall is high, suggesting that the model can accurately detect a large proportion of the parking slots in the images. The F1-Confidence Curve is a combined measure of precision and recall, offering a balanced evaluation by considering both false positives and false nega- tives. The F1 score provides an overall assessment of the model's capability to accurately predict parking slots under varying confidence levels. As seen in the plot, the model achieves a high F1 score, demonstrating that it maintains an optimal balance between precision and recall. The Precision-Recall Curve further quantifies the trade-off between precision and recall, allowing for a visual comparison of the model's performance. The curve clearly indicates that the model achieves a high level of performance for both the "parked" and "vacant" classes, as indicated by the respective precision and recall values. The mAP value suggests that the model exhibits excellent precision and recall at various confidence thresholds, making it well-suited for real-time car parking slot detection. Additionally, the Confusion Ma- trix (see Fig. 10) provides a detailed breakdown of the model's classification results, showing the true positives, false positives, true negatives, and false negatives. In the matrix, the rows represent the actual classes (parked, vacant, and background), while the columns represent the predicted classes. The values in the matrix show the proportion of cor- rect and incorrect classifications made by the model. The values on the diagonal indicate correct classifications, with 99% accuracy for both parked and vacant classes. The off-diagonal values indicate misclassifications, with a small percentage of false positives for "vacant" slots and false negatives

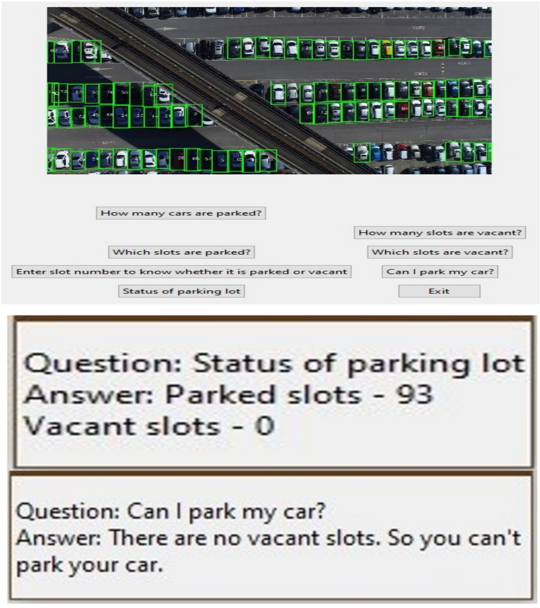

Figure 8: Images of GUI for user interaction selecting different questions from the menu

for ”parked” slots. This detailed analysis helps in identifying specific areas where the model can be improved, such as minimizing misclassifications and ensuring better handling of background noise. Together, these evaluation metrics offer a comprehensive understanding of the model’s performance, ensuring that the car parking slot detection system is both accurate and reliable across various conditions.

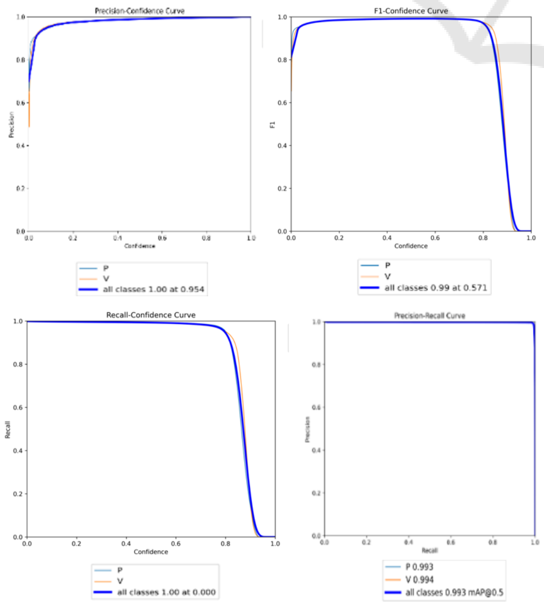

Figure 9: Precision-Confidence, F1-Confidence, Recall-Confidence and Precision-Recall Curve

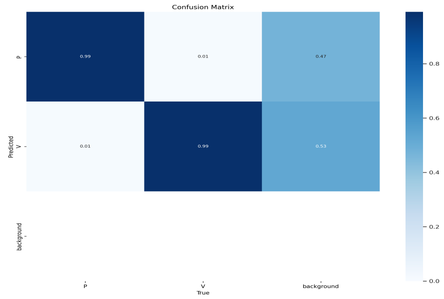

Figure 10: Confusion Matrix

3.2 Limitations

The model’s performance in detecting parking slots raises several critical considerations, particularly in environ- ments where the parking area is surrounded by or covered by trees, as trees can obstruct camera views, complicating overhead monitoring. Cameras on tall buildings may struggle with clear images. Proper camera placement is cru- cial, as low angles can miss sections of the lot, while high angles may hinder depth perception, making it difficult to distinguish between closely parked cars and vacant slots. These factors significantly impact detection accuracy.

The challenge is illustrated in Fig. 11, where the model struggles to accurately predict all parked and vacant slots due to the dataset’s capture angle. The camera’s position significantly affects prediction accuracy. To address this, training the model on a more diverse dataset with images from various angles and perspectives can enhance its generalization and robustness. Exposure to different viewpoints during training will enable the model to adapt to variations in camera positions, improving its ability to predict parking slot occupancy in real-world scenarios. Ultimately, diverse data collection is crucial for optimizing performance across varying conditions.

Figure 11: Image showing improper predictions

4 CONCLUSION

In this study, we proposed a robust real-time VQA model for car parking slot detection, utilizing the state-of-the- art YOLOv8 object detection framework. The model successfully detected and classified parked and vacant parking slots with an impressive mAP of 0.993 at an IoU of 0.5. Through the application of the PKLot dataset, the research demonstrated the critical importance of both high-quality and sufficiently diverse datasets, alongside effective preprocessing techniques, in achieving optimal model performance. The integration of a menu-based question-answering interface further enhanced the usability of the system, allowing for interactive user queries. Our results emphasize that YOLOv8 is capable of delivering accurate and reliable results even in scenarios with limited data. The results under- score that both the quality and quantity of the dataset, along with the chosen processing techniques, are key factors for the model's success. These findings validate the robustness of the YOLOv8 model across varied conditions, show- ing its capability to deliver precise and reliable car parking analysis in real-time applications. The metrics, including precision, recall, and F1-score, collectively highlight the efficiency and potential of this model for real-time parking lot monitoring. Future work will aim at further enhancing model accuracy, exploring the integration of additional features, and optimizing the system for scalability across larger datasets and varied environmental conditions.

4.1 Future scope

For future work, the proposed car parking slot detection system could be expanded in several ways to enhance its applicability and performance. Firstly, the system could be improved by integrating real-time video processing to track vehicle movement dynamically, allowing for real-time parking slot updates. This would enable the system to monitor multiple camera feeds simultaneously in large parking areas, ensuring up-to-the-minute information on slot availability. Additionally, the model can be further optimized to handle more challenging conditions such as adverse weather (rain, fog, snow) by incorporating specialized datasets for these conditions. Advanced techniques like multi- sensor fusion, which combines data from cameras, LiDAR, or radar, can be explored to enhance detection in areas where visibility is limited. The integration of cloud-based solutions for centralized parking lot management could also facilitate easier scalability and allow for larger-scale deployment across various environments, including urban and rural settings. Another advancement is creating a model that can identify free parking spaces without relying on visible parking lines. This capability would enhance the system's versatility, allowing it to function in environments like regular streets or unmarked lots, thereby expanding its applicability to various real-world scenarios. Lastly, the system could be adapted to incorporate payment and reservation features for users, providing a seamless end-to-end solution for parking lot management. The addition of predictive analytics and AI-driven recommendations, which predict future slot availability based on usage patterns, could also be an interesting avenue for future research and development. This would make the system more intelligent and user-friendly, offering a holistic approach to smart parking solutions. By addressing these areas, the system could evolve into a more robust, accurate, and adaptable solution for smart parking management.

REFERENCES

Lu, Siyu & Ding, Yueming & Liu, Mingzhe & Yin, Zhengtong & Yin, Lirong & Zheng, Wenfeng. (2023). Multi-scale Feature Extraction and Fusion of Image and Text in VQA. International Journal of Computational Intelligence Systems. 16. 10.1007/s44196-023- 00233-6.

Hela, Vikash & Arakkal, Lubna & Kalady, Saidalavi. (2022). CarParkingVQA: Visual Question Answering application on Car parking occupancy detection. 1-6. 10.1109/INDICON56171.2022.10039925

Ding, Xiangwu & Yang, Ruidi. (2019). Vehicle and Parking Space Detection Based on Improved YOLO Network Model. Journal of Physics: Conference Series. 1325. 012084. 10.1088/1742-6596/1325/1/012084

Li, Borui. (2022). Detection of Parking Spaces in Open Environments with Low Light and Severe Weather. 10.2991/978-2-494069-31-2_354

Zhang, Xin, Wen Zhao, and Yueqiu Jiang. "Study on Parking Space Recognition Based on Improved Image Equalization and YOLOv5." Electronics 12.15 (2023): 3374.

Sairam, Bandi, et al. "Automated vehicle parking slot detection system using deep learning." 2020 Fourth international conference on computing methodologies and communication (ICCMC). IEEE, 2020.

Singh, C., & Christoforou, C. (2021). Detection Of Vacant Parking Spaces Through The Use Of Convolutional Neural Network. The International FLAIRS Conference Proceedings, 34. https://doi.org/10.32473/flairs.v34i1.128470

Huang, Shan & he, ye & Chen, Xiao-an. (2021). M-YOLO: A Nighttime Vehicle Detection Method Combining Mobilenet v2 and YOLO v3. Journal of Physics: Conference Series. 1883. 012094. 10.1088/1742-6596/1883/1/012094

Amato, Giuseppe & Carrara, Fabio & Falchi, Fabrizio & Gennaro, Claudio & Vairo, Claudio. (2016). Car parking occupancy detection using smart camera networks and Deep Learning. 1212-1217. 10.1109/ISCC.2016.7543901

Nguyen, Duy-Linh & Vo, Xuan-Thuy & Priadana, Adri & Jo, Kang-Hyun. (2023). YOLO5PKLot: A Parking Lot Detection Network Based on Improved YOLOv5 for Smart Parking Management System. 10.1007/978-981-99-4914-4_8

P. Sharmila, P. Rohinth, G. Sarvesh, P. Priyadharshan, M. Balasakthishwaran and S. Gandhi, "Next-Gen Parking Solutions," 2024 International Conference on Communication, Computing and Internet of Things (IC3IoT), Chennai, India, 2024, pp. 1-6, doi: 10.1109/IC3IoT60841.2024.10550402.

V. Naik, V. Borkar, K. M. C. Kumar and P. Shetgaonkar, "A Survey On Various Smart Car Parking Systems," 2024 3rd International Conference for Innovation in Technology (INOCON), Bangalore, India, 2024, pp. 1-6, doi: 10.1109/INOCON60754.2024.10511587.

D. P. Surve, K. Shirsat and S. Annappanavar, ""Parkify" Automated Parking System Using Machine Learning," 2023 International Conference on Integrated Intelligence and Communication Systems (ICIICS), Kalaburagi, India, 2023, pp. 1-7, doi: 10.1109/ICIICS59993.2023.10421723.

D. Balusamy, G. Shanmugam, B. Rameshkumar, A. Palanisamy, G. Kingston and L. Ravi, "Autonomous Parking Space Detection for Electric Vehicles Based on Advanced Custom YOLOv5," 2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT), Vellore, India, 2024, pp. 1-5, doi: 10.1109/AIIoT58432.2024.10574648.

V. Shankar, V. Singh, V. Choudhary, S. Agrawal and B. Awadhiya, "An Enhanced Car Parking Detection Using YOLOv8 Deep Learning Capabilities," 2024 First International Conference on Electronics, Communication and Signal Processing (ICECSP), New Delhi, India, 2024, pp. 1-5, doi: 10.1109/ICECSP61809.2024.10698155.

Sarhan, A., Abdel-Rahem, R., Darwish, B. et al. Egyptian car plate recognition based on YOLOv8, Easy-OCR, and CNN. Journal of Electrical Systems and Inf Technol 11, 32 (2024). https://doi.org/10.1186/s43067-024-00156-y

P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001, Kauai, HI, USA, 2001, pp. I-I, doi: 10.1109/CVPR.2001.990517.

N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1, doi: 10.1109/CVPR.2005.177.

Piccinini, Paolo, Andrea Prati, and Rita Cucchiara. "Real-time object detection and localization with SIFT-based clustering." Image and Vision Computing 30.8 (2012): 573-587.

R. Gandhi, "R-CNN, Fast R-CNN, Faster R-CNN, YOLO — Object Detection Algorithms," 2018.

Xiangwu Ding and Ruidi Yang, "Vehicle and Parking Space Detection Based on Improved YOLO Network," Conf. Ser. 1325 012084