SEMARAK ILMU PUBLISHING
202103268166(003316878-P)

Journal of Advanced Research in Applied Sciences and Engineering Technology

Journal homepage:
https://semarakilmu.com.my/journals/index.php/applied_sciences_eng_tech/index
ISSN: 2462-1943

Parking Slot Detection and Vacancy Check Based on Deep Learning Method

Fahadul Islam[1], Md. Shohel Arman[1], Ong Bi Lynn[2,3], Hasnur Jahan[1], R. Badlishah Ahmad[2,3], Naimah Yaakob[2,3], Nur Farhan Kahar[2,3], Md. Maruf Hassan[1,2,*]

1 Department of Software Engineering, DIU Data Science Lab, Daffodil International University, Daffodil Smart City, Birulia 1216, Bangladesh
2 Department Faculty of Electronic Engineering & Technology, Universiti Malaysia Perlis, 02600 Arau, Perlis, Malaysia
3 Centre of Excellence for Advanced Computing (ADVCOMP), Universiti Malaysia Perlis, 02600 Arau, Perlis, Malaysia

ABSTRACT

In Bangladesh, traffic is a huge problem, mostly in urban areas like Dhaka. Parking at the roadside is a major problem for traffic in Bangladesh. Private transport user waste most of their time in their daily life searching for a parking slot. Therefore, there is a crucial need to develop a system to help people find vacant parking spots. Systems for managing parking spaces, such as those that identify empty spaces, can help minimize traffic and energy waste in big cities. Since visual approaches may use security cameras installed already in several parking lots, they are an affordable alternative to other methods of vacancy detection. Based on the deep learning model, we have made a cost-effective system and will apply data from that camera. After applying different deep-learning models, we implemented YOLOv7, Mask-RCNN, and YOLOv5. Class loss, Box loss, instances, mAp, and Object loss are all generated by the model. Among all models, YOLOv5 has the highest mAp, 98%. In the future, we will work on providing access to a registered user in the parking garage. This will increase the dataset and obtain higher accuracy.

Keywords:

Parking; YOLO; Deep learning; Detection

1. Introduction

Every day, more and more people are becoming reliable on transport. Transportation is a big part of how people live in the modern world, which explains why more individuals purchase automobiles. Consequently, the number of total vehicles is exponentially increasing [1]. With more and more cars on the road, finding a parking space is becoming a problem worldwide. In addition, finding a parking space in a large city with many people who work for the owner or the driver themselves is becoming increasingly difficult [4]. If one cannot find a place to park in the area he wants, it can cause much traffic [18]. Normally, a person only uses their hands to control steering to park vehicles. People also get upset when they are unable to find a parking spot for a long time. However, many of the parking lots may have empty spots that drivers in the area do not know about. Therefore, the automatic

* *Corresponding author.*
E-mail address: ancssf@gmail.com

https://doi.org/10.37934/araset.62.1.4966

parking system is a new area that researchers in computer vision are interested in getting involved with [6].

The already-created systems mostly rely on machine learning or picture segmentation. Neural Networks (NN) and Support Vector Machines (SVM) over spot patches [17].

In recent articles and shows, both TV and media have discussed environmental pollution caused by traffic, road congestion, and traffic monitoring systems. They discussed issues like road accidents, vehicle rules, which are odd-even in Dhaka, and fuel and diesel price rises [9]. Many researchers are developing systems that survey on parking spots in real-time since they do not require a special setup, and sensor-based computer parking systems that employ security cameras to do the same task are less expensive [19]. There is no need to set up these vision-based systems any further. For intelligent city parking systems to work, parking spots need to be planned based on how they are being used. Notably, parking spaces cannot be reserved in real life the same way a movie ticket can. We have taken note of this problem and devised a way to use the Statistical Block Matching Algorithm (SBMA) for accessible, real-time recorded video. In the literature review, different ways to figure out how blocks are moving are described, along with research gaps. The technique of block-matching and the suggested mechanism were highlighted for supplying real-time parking statuses [21]. The simulation results of the proposed method are provided, along with an analysis of how well it works based on F1 scores, recall, and precision.

The deployment of real-time traffic management systems in Dhaka is varied compared to the widespread application of comparable systems outside of Bangladesh. Giving users the most recent information on parking spot availability is one of the ITS's (Labtest Bangladesh Ltd) crucial tasks. We originally developed real-time parking space availability in a tiny module within a toy car and different lighting situations using circular Hough transforms [23]. We also developed an intelligent parking system. The idea of finding parking spots was inspired by real-time movie theatre ticket ordering systems. In big cities, parking capacity and volume statistics vary greatly, notably in neighbourhoods near commercial districts, theatre districts, medical facilities, and expansive malls. When drivers have access to parking information in advance, they can save time, money, and gasoline while reducing traffic congestion [10]. Today, people who research, generate employment and amass riches harbour ambitions to launch their own businesses and become entrepreneurs. Therefore, it is becoming increasingly evident that patents need to be registered.

One of these applications that has a patent is the automated parking module. The case of campus study in Tezcan provides an overview of a parking fee module developed as a traffic demand management technique to reduce the usage of private automobiles and minimize road congestion for academics and residents, which is even. This was completed (2012) in order to fulfil the demands of the community as a whole [14].

The people's number and automobiles entering cities has expanded along with the expansion of the economy and the migration of people from more rural to more urban regions [15]. In one of the biggest cities on the planet, most motorists are irritated when attempting to locate a parking space.

Dhaka City was rated as the second hardest city to find parking, as reported by IBM, according to a poll that was conducted since it takes a long time to find parking, and it is never guaranteed that there will be any available. Finding a parking spot takes time, which is bad for the person looking for a spot and the city's general traffic flow.

However, nothing has been put forth. Moreover, some solutions have been proposed as a solution to the parking issue. Thus, none has proven effective. One of these ideas is to increase parking fees in malls and localities. The public believed that this was an effort to force them to use public transit and, as a result, leave their cars at home.

It is feasible that adding extra parking spaces to the city might help solve the parking space issue. However, this solution is impractical and will not work as a city grows. Therefore, computer vision and deep learning technology may be used to alert drivers on where to find free parking spots in the city. This will assist in solving the parking problem that blights our towns and end the parking crisis. This study focused on investigating the application of computer vision technologies, constructing an effective machine learning model in parking space detection under varied lighting settings and weather conditions, and solving the parking crisis. These goals were the only ones covered under this study's scope.

2. Literature review

To work properly, we have reviewed numerous literatures and reviewed them. Valipour *et al.,* [2] proposed a vision-based detection method that allows a single camera to monitor multiple booths, thereby reducing the cost of the system per stall. The detection procedure improved the AUC(area under the ROC Curve) by 8.13%. It also demonstrated performance, which is robust in various testing situations, along with public camera evaluations. The large Federal University of Parana and the Pontifical Catholic University of Parana parking sites are quite costly.

ResNet18 was suggested by Valipour *et al.,* [3] and was discovered to perform better than other detectors, such as You Only Look Once (YOLO), YOLO-conv, GoogleNet, and ResNet50. Axis Q1942-E thermal camera data was used to gather vehicle detection information. The thermal camera was positioned on two-story buildings and has a focus length of 19 mm. It is normal for a car to park between two spots due to the potential of parking lines being hidden by snow.

Mahmud *et al.,* [4] proposition vacant parking space detection, and this study highlighted the advantages and drawbacks of the currently used techniques. CNRpark and Roboflow can only function in indoor parking environments with a number of support vehicles, according to the Pk-Lot Dataset CNN Segmentation, Object Detection, NN, and Parking Space Management.

Amato *et al.,* [5] posited that Convolutional Neural Networks (CNN) should be used to solve the provided problem effectively. In particular, it demonstrates extremely high accuracy even in the face of noise caused by changing lighting conditions, shadows, and partial occlusions. KPLot and CNRpark include Convolutional Neural Networks(CNN), deep learning, classification, and machine learning.

Another study employed a deep CNN and binary SVM classifier to determine the occupancy of outdoor parking spots from photos. Acharya *et al.,* [6] established a robust parking occupancy identification system. The parking spots were collected by PKLot, Barry Street, which also caught the bias in the training data utilized by CNN and SVM, as well as shadows cast by buildings on spaces and high sun reflection from automobiles.

Using Detectron2 and deep learning, Šćekić *et al.,* [7] demonstrated a method for categorizing parking spots from parking lot photos PKLot into two categories: vacant and occupied. As can be observed, few vehicles are visible in these shots, which was to be expected considering that they were taken with a completely different camera type, perspective, and location. These images need to be improved, labelled, and contributed to a larger dataset to produce a more accurate prediction model.

Jiang an *et al.,* [9] in this study used traditional image processing techniques to detect parking slot markings. It demonstrated basic feasibility but struggled with varying lighting conditions and slot types. The method was not robust enough to handle complex illumination or different parking slot orientations. This study leverages the Mask R-CNN algorithm, which provides superior robustness to varying lighting conditions and can accurately detect parking slots in diverse orientations, thus overcoming the limitations of traditional techniques.

Amato and *et al.,* [17] in this study explored the use of traditional image processing techniques for parking lot occupancy detection. Although it demonstrated promising results, it did not leverage deep learning techniques, which have shown superior performance in various vision tasks. Their study advances this work by utilizing a deep CNN specifically designed for smart cameras, addressing the limitations of traditional methods and enhancing detection accuracy.

Lin and *et al.,* [18] in this study implemented traditional image processing techniques for traffic flow counting, demonstrating the basic feasibility of using image recognition in ITS. The approach struggled with dynamic and complex traffic conditions, leading to inaccuracies in vehicle counting. This study advances this work by employing the YOLO framework, which is more robust and accurate in detecting vehicles in complex environments.

Zinelli and *et al.,* [24] in this study utilized static cameras to detect parking slots, achieving reasonable accuracy in controlled environments. The reliance on static cameras limited the system's applicability in dynamic driving scenarios and reduced its robustness to varying conditions. This study uses surround-view images generated by fusing views from four different cameras mounted on a vehicle, providing a comprehensive and dynamic perspective that overcomes the limitations of static camera-based systems.

Ichihashi and *et al.,* [26] in this study explored sensor-based techniques for parking lot vacancy detection, demonstrating reliable performance in both indoor and outdoor environments. Despite their reliability, sensor-based systems can be costly to install and maintain, and they often lack the flexibility of camera-based systems. This study aims to provide a cost-effective and flexible alternative by developing a camera-based system that can operate effectively in outdoor environments, thus addressing the limitations of sensor-based approaches.

3. Methodology

3.1 Data Collection

A wide range of studies have contributed to our understanding of parking-related issues, and the landscape of publicly available parking statistics on the Internet is vast and varied. The "pklot" dataset, the "car pk" dataset, and the CNR parking dataset are notable examples that jointly offer insights into parking dynamics. These statistics, collected from different parts of the world, shed light on parking situations, primarily in outdoor settings throughout Europe and beyond.

Despite the abundance of parking statistics accessible, a notable gap has been noticed: there are no databases specifically dedicated to parking garages in Bangladesh. To fill this gap in the parking dataset landscape, a fresh project has been launched. Moreover, an initiative to gather data exclusively from parking garages in Dhaka, Bangladesh, was established to recognize the potential impact and applicability of such data.

Being a busy metropolis, Dhaka has an intricate parking ecosystem built into its urban design. The foundation of this project is made up of datasets obtained from a Dhaka parking garage. These statistics provide a distinctive viewpoint conspicuously lacking in currently accessible datasets by capturing the delicate interplay between automobiles, available spots, and parking dynamics in a garage setting.

The study seeks to fill a research gap by starting this data collection journey within the setting of parking garages in Bangladesh. The collected datasets are quite useful for regional and international research projects due to the distinctive characteristics of parking behaviours and difficulties encountered in this situation. In addition to enhancing parking-related datasets, this endeavour makes it easier to investigate parking optimization tactics, mobility analytics, and smart urban planning solutions adapted to Dhaka's unique circumstances and similar urban environments.

In essence, the selection of Bangladesh's Dhaka parking garage statistics to be curated fills a critical gap in the availability of parking data sources. These databases offer a platform for cutting-edge research and data-driven insights, enabling a deeper understanding of the complex parking dynamics within bustling urban environments like Dhaka as cities worldwide struggle with parking congestion and management.

The Dataset Preprocessing Program has been painstakingly created to handle a dataset comprising 18 photos taken by a CCTV camera in the context of a car garage. Each photo captures a unique view of the garage's interior, displaying two parked automobiles tucked away on the property and the watchful presence of a security guard. The dataset offers a thorough visual record of the automobile garage, providing information on the positioning of the vehicles and the security officers, all of which were photographed using surveillance technology.

A number of critical procedures are smoothly conducted by the program to preprocess this dataset. In order to ensure that all 18 images are properly imported into the program's memory, the program begins by importing the raw image data. In order to properly emphasize the key items of interest—the parked automobiles and the security guard—the computer then uses sophisticated image cropping algorithms to remove superfluous background components. Additionally, the program carefully recognizes and delineates the parked automobiles and the security officer in each image, demarcating them with precise bounding boxes using object identification algorithms.

The dataset's potential for various applications, including object detection, activity analysis, and scene comprehension in the context of a car garage setting, can only be fully realized with the help of our preprocessing tool. The carefully selected photos, which draw attention to the presence of vehicles and people, serve as a useful tool for creating and honing complex models in the fields of computer vision and surveillance analytics.

3.2 Data Preprocessing

The first stages of data preparation were initiated after data from various scenarios within a parking garage in Dhaka had been successfully collected. The gathered dataset serves as the starting point for training a machine-learning model. It was painstakingly assembled to mimic real-world parking dynamics accurately. The precise preparation of the dataset is the first crucial step, which enables the computational model to be easily learned.

The creation of solid machine learning is essential to accomplishing this project. The dataset's capacity to provide the model accuracy and insights increases dramatically as preprocessing continues. This sequence illustrates the transition from unrefined information to refined knowledge, which is the basis for efficient model training.

Data used to start this voyage came from a cross-section of an old parking garage in Dhaka, which provided a thorough picture of actual conditions (see Figure 1). A clear separation between two different road groups was discovered in this dataset, serving as the foundation for further classification. A training set and a validation set were created using a careful method of dataset partition, with 80% of data points used in training while the remaining 20% used in testing.

Accordingly, 1,600 observations that comprise the training dataset were collected during various hours, with 800 being recorded during the day and an equal number at night. The temporal fluctuation is similarly represented in the test dataset's 600 observations, which includes 300 observations taken during the day and 300 observations taken at night.

Fig. 1. The dataset consists of 18 pictures of a car garage photo taken using a CCTV camera where the picture shows pictures of 2 cars parked inside of it along with a guard

Data standardization was essential for ensuring consistency and comparability. Pixel values were rescaled to fall between 0 and 225 using min-max normalization (see Figure 2). This normalizing procedure promotes homogeneity and enhances the model's performance in a range of lighting and settings conditions.

Undoubtedly, data scaling has become a major obstacle in image processing, with the goal of improving model effectiveness. The dataset was painstakingly scaled into a standard format of (224, 224) pixels, facilitating further processing and analysis (Figure 2).

In conclusion, the trajectory from data collection in a parking garage in Dhaka to the initial steps of preprocessing reveals a thorough and organized strategy. Data categorization, division, standardization, and resizing are all steps that lead to a carefully curated dataset ready to provide the machine-learning model with the ability to understand complex parking dynamics. The deliberate retention of critical numbers that demonstrate data sources, normalization, and resizing highlights the thorough and comprehensive character of the technique.

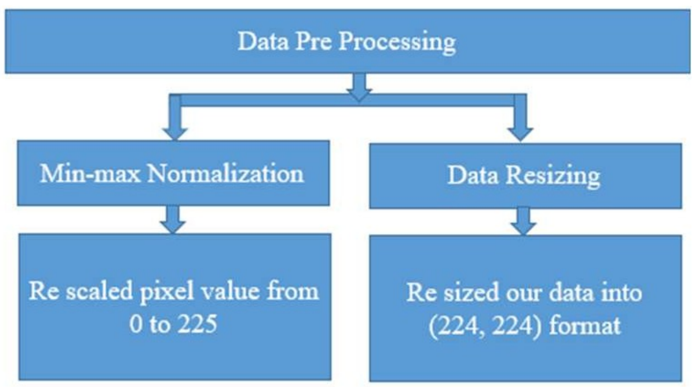

Fig. 2. Dataset preprocessing classification into Min-max Normalization that has a sub-division called rescaled pixel value, and another part of classification is Data Resizing, which is extended into resizing the data format for our data

3.3 YOLOv5

The YOLO model, with its groundbreaking method, is recognized as a groundbreaking achievement in the field of computer vision. In essence, YOLO departs from conventional approaches that include multi-stage pipelines and re-imagines object detection as an end-to-end procedure. The key feature of YOLO is its capacity to process a whole image in a single forward pass through the network, which gives it exceptional efficiency and real-time application [16].

YOLO radically changes the architecture in contrast to its predecessors, which use area proposal networks followed by classification and refining stages. It creates a grid out of the input image, with each grid cell being in charge of determining whether any objects are present. Since each cell directly predicts bounding boxes and class probabilities, this method greatly speeds the process. Additionally, this grid-based approach speeds up detection and enables YOLO to cogently represent spatial relationships between items.

The two primary parts of the YOLO architecture are the backbone network, which analyses the image and extracts its features, and the detection head, which produces predictions based on these data. The detection head at the network's end forecasts item bounding boxes and corresponding class probabilities. In addition to enhancing YOLO's real-time capabilities, this integrated architecture encourages end-to-end learning, allowing the model to improve its feature extraction and prediction simultaneously.

In the YOLO training procedure, a composite loss functions that accounts for localization loss, classification loss, and confidence loss is optimized. The localization loss penalizes inaccurate bounding box predictions, resulting in precise object localization. In contrast to the confidence loss, which indicates the level of trust in object predictions, the classification loss addresses the proper classification of objects within boxes. Notably, YOLO uses a single-stage training technique that optimizes every part at once to improve the coherence and effectiveness of the model.

One of YOLO's advantages is its ability to recognize items of different scales and sizes inside the same image, resulting from the grid-based methodology. Due to the set cell sizes, YOLO naturally has trouble identifying small objects. However, later versions, such as YOLOv3 and YOLOv4, added feature pyramids and anchor boxes to address this issue. These developments allow YOLO to adapt to a range of object sizes and complexities successfully.

Non-Maximum Suppression (NMS) is a crucial component of YOLO's post-processing. In regions of overlapping detection, NMS filters out duplicate bounding boxes by choosing the one with the highest confidence score once predictions have been created. This process eliminates redundant predictions, resulting in cleaner outputs and more accuracy in general.

Applications for YOLO can be discovered in a wide range of industries, including robotics, autonomous vehicles, surveillance, and healthcare. In situations when swift decision-making is necessary based on detected objects, its capacity to quickly process photos and offer real-time results is crucial. Despite YOLO excels in real-time detection, there are trade-offs; in some complicated scenarios, it may not equal the accuracy of multi-stage detectors.

In conclusion, the grid-based, end-to-end architecture of the model YOLO has completely changed how objects are detected while providing unmatched efficiency and real-time capabilities. Together with advancements like anchor boxes and NMS, its ability to process full images in a single pass exemplifies YOLO's ongoing evolution to address problems and stay relevant in the fast-paced field of computer vision.

3.4 Evaluation Methods

A confusion matrix has been built to examine the data. For assessment, it is necessary to know the true and negative, true and positive, false and negative values, and false and positive. A true positive in this context is a real number that was correctly predicted. A precise negative result is disregarded. When a positive value is incorrectly inferred, the prediction is considered false-positive. A false negative result is referred to as a "false negative" (Figure 3). This confusion matrix has a car, a parking space, and a background in its x-axis. The confusion matrix for the same class is 1, as well as the confusion matrix for the car and background.

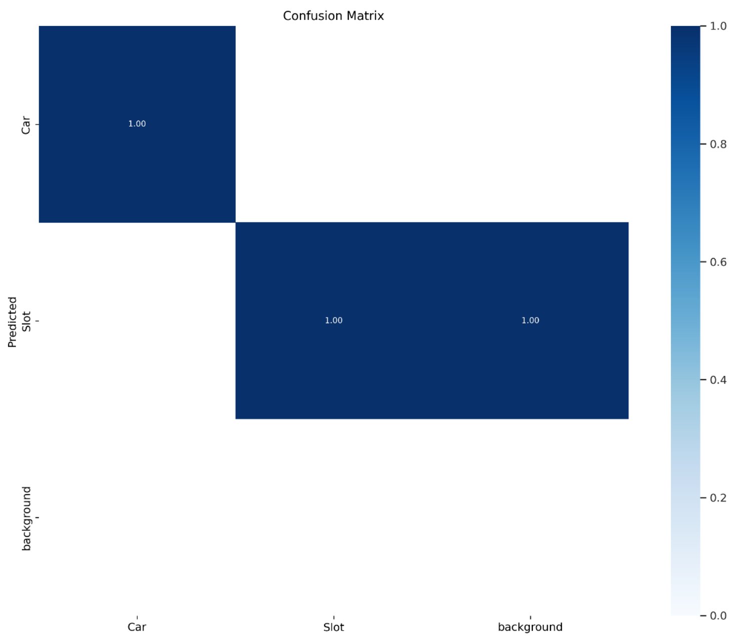

Fig. 3. Confusion Matrix to analyse data

3.5 Accuracy

In the field of machine learning, accuracy is a crucial cornerstone that serves as the primary performance indicator for a model. By comparing the model's output with actual data, it indicates the model's ability to generate precise predictions. Thus, accuracy is a goal that extends beyond the classroom and has significant implications for practical use.

Accurate forecasts can result in early disease diagnosis in industries like healthcare, potentially saving lives. Exact models guide investment choices and risk evaluations in finance. Accuracy, however, is a hard balance to strike. Overly complicated models may perform well on training data due to overfitting but poorly on new information.

Class imbalances within datasets can also deceive accuracy evaluations. Notably, a high accuracy score may be deceiving if a model only forecasts the majority class when predicting a rare event. The pursuit of accuracy continues to be a driving premise in machine learning. However, it is moderated by ideas like generalization, interpretability, and real-world application. The accuracy equation is depicted in Figure 1. The sum of true and positive value, and true and negative value is divided by the sum of the true and negative value, the true and positive value, the False and positive value, and the false and negative value in this instance:

$$\text{Accuracy} = \frac{True\ Positive + True\ Negative}{True\ Positive + False\ Positive + True\ Negative + False\ Negative}. \tag{1}$$

3.6 Precision

The assessment of a model's effectiveness in the complex field of machine learning heavily depends on a statistic called "precision." This metric is crucial for accurately and dependably determining how well the model performs its intended purpose. We use precision as a compass to navigate the maze of predictions and assess the model's ability to properly identify successful outcomes in the sea of data.

Precision is determined using a complex approach that reveals the model's actual performance. A quantitative evaluation of precision is obtained by considering the instances that the model labels as positive and contrasting them with the instances that are truly positive. The equation accuracy = true positives/(true positives + false positives) captures this comparison. It represents a fine balance between genuine positives, which are cases that are accurately classified as positive, and false positives, which are situations that are incorrectly classified as positive.

The importance of precision can be observed in various applications, such as fraud detection and medical diagnosis. Precision, for example, describes a model's capacity to precisely identify abnormalities, reducing the possibility of a wrong diagnosis in medical imaging. Similarly, accuracy is the cornerstone of effective fraud detection systems in the financial world, preventing inaccurate classifications and monetary losses.

The pursuit of precision, meanwhile, is not without difficulties. It takes skill to strike the correct balance between true positives and false positives since overly cautious models may boost precision by classifying fewer instances as positive but at the expense of missing potentially useful insights.

Therefore, when evaluating the effectiveness of a machine learning model, precision becomes a crucial criterion. Its computation captures the essence of a model's predictive accuracy and its capacity to separate signal from noise. Precision, which enables us to make educated decisions through its complex calculation and interpretation, walks a tight line between caution and exploration while maximizing the potential of machine learning in various fields. Figure 2 illustrates the precision value equation, which divides the true and positive values by the sum of the true and positive values and the false and positive values:

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive} = \frac{True\ Positive}{Total\ Predicted\ Positive}. \tag{2}$$

Figure 4 depicts a precision-recall diagram, with precision on the y-axis and recall on the x-axis. A specific threshold is applied to the output probabilities or scores of the model at each point on the curve. The model's classification choices likewise change as the threshold does, affecting recall and precision.

The graph itself illustrates how precision and recall progress as the threshold changes. In general, precision rises as the threshold is tightened, the model is more careful in categorizing positive cases, and there are fewer false positives. This, however, frequently comes at the expense of recollection, missing some actual good examples and lowering recall.

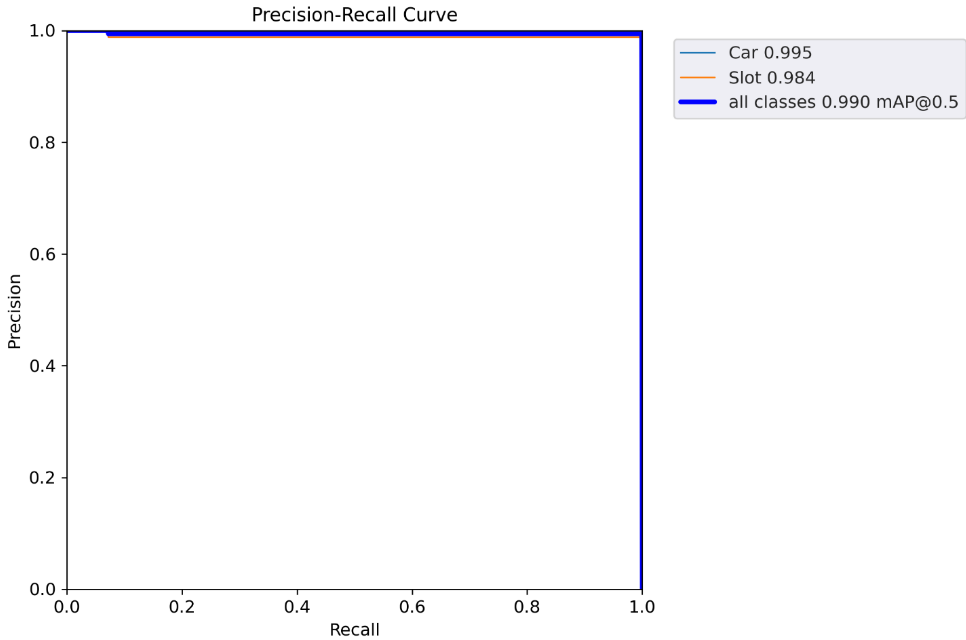

Fig. 4. Precision-Recall curve

The precision-recall curve's shape offers essential clues about the overall performance of the model. The model's capacity to correctly categorize positive examples while reducing false positives is demonstrated by a curve that hugs the upper-right corner and demonstrates a solid balance between precision and recall. On the other hand, a downward-deviating curve indicates a trade-off where one metric advances at the expense of the other.

Additionally, the precision-recall area curve is sometimes used to calculate a single statistic known as the Average Precision (AP). This statistic measures the model's overall performance across several thresholds and clearly illustrates its capacity for high precision and acceptable recall rates.

In the figure, we can observe that the precision-recall value for the car is 0.995, and the precision value for the parking slot is 0.984.

3.7 Recall

Recall, sometimes referred to as sensitivity or true positive rate, is an essential performance parameter in the field of machine learning that measures a model's capacity to accurately identify positive examples from the whole population of real positive instances within a dataset. Furthermore, recall essentially assesses how well a model captures all relevant instances of a given class, making it particularly crucial in situations were failing to identify positive cases has major repercussions. Recall is mathematically determined by dividing the total of true positives and false negatives by the number of true positive predictions. Meanwhile, true and positive are occurrences that the model correctly classifies as positive, whereas false and negatives are situations that the model wrongly classifies as positive but are actually positive.

With a high recall value, the model effectively reduces false negatives and ensures that very few real positive cases are missed. This is particularly crucial in applications like medical diagnosis or fraud detection when the cost of missing a positive occurrence is considerable. For example, a high recall in medical imaging would indicate that the model is efficient in spotting probable diseases or anomalies, lowering the likelihood of misdiagnosis.

High recall, meanwhile, can come at the sacrifice of precision, another crucial parameter. Hence, achieving a high recall typically requires being more accurate in classifying cases as positive, which could result in a higher number of false positives. Precision focuses on the fraction of real positive predictions among all positive predictions.

Recall and precision are frequently trade-offs. Correspondingly, finding the ideal balance depends on the precise goals of the machine learning activity. The decision-making process is occasionally guided by domain expertise or the potential effects of false negatives and false positives.

In conclusion, recall measures a model's capacity to identify positive cases thoroughly. It is essential in situations when the expense of omitting affirmative cases is high. Machine learning models can be improved to produce decisions consistent with the desired result by balancing recall, precision, and the task's requirements.

The equation states that the true positive value, which is the sum of the true positive and false negative values, is subtracted from the total positive real value to determine the recall value:

$$\text{Recall}=\frac{True\ Positive}{True\ Positive+False\ Negative}=\frac{True\ Positive}{Total\ Actual\ Positive}. \qquad (3)$$

The relationship between the recall (sensitivity or true positive rate) and the confidence thresholds of a classification model's predictions is visualized using a recall-confidence diagram, also referred to as a recall-precision curve or a precision-recall curve. The model's recall and precision performance are displayed in this graphic as a function of the confidence threshold.

The recall-confidence diagram's x-axis, which runs from the lowest to the greatest confidence values, exhibits various confidence thresholds that are applied to the model's predictions. The relevant recall values at each threshold are displayed on the y-axis. The model tends to grow more cautious and make less accurate predictions as the threshold rises. Typically, this results in increased precision but possibly lower recall.

The trade-off between recall and precision as the confidence threshold changes is depicted by the curve on the recall-confidence diagram. Lower thresholds cause the model to identify more positive cases, increasing recall but perhaps reducing precision due to increased false positives. Recall may decline as the threshold increases. However, precision often increases as the model gets more selective about which occurrences to categorize as positive.

This visual aid makes it easier to choose the ideal confidence level that best satisfies the task's needs for precision and memory. Similar to calculating the AP in a precision-recall curve, the area under the recall-confidence curve can be used as a general performance indicator.

Recall-confidence diagrams are particularly helpful when working with unbalanced datasets or when the cost of false positives and false negatives vary. Considering the application's objectives helps decision-making on the threshold that maximizes the desired result. Overall, this graphical depiction helps users optimize model behaviour and determine the ideal balance between recall and precision for a particular activity. From Figure 5, the recall value of all classes is 0 at confidence level 1.

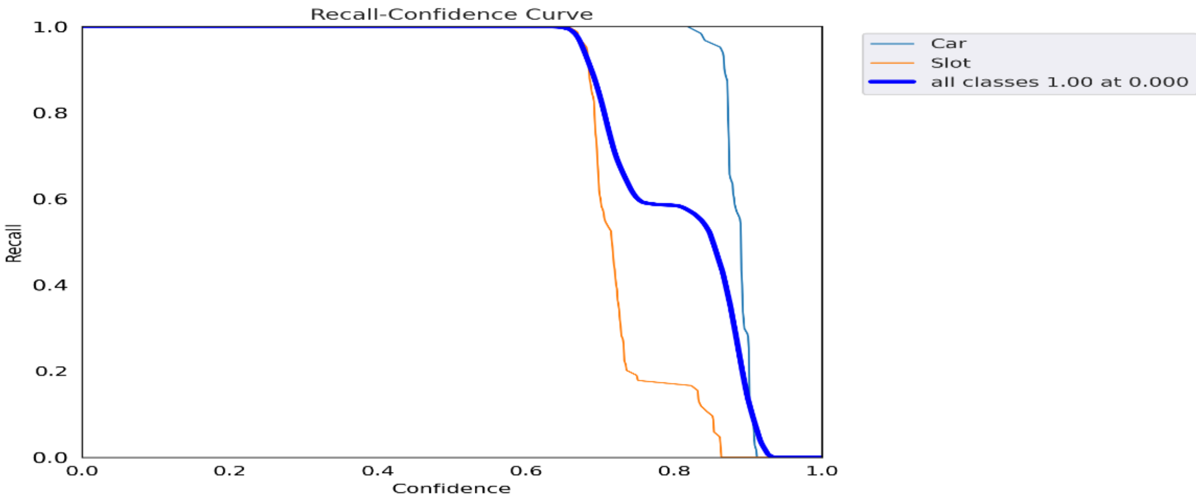

Fig. 5. Recall-Confidence curve

3.8 F1 score

A common performance statistic in machine learning is the F1 score, which combines recall and precision into a single value to provide a fair assessment of a model's general accuracy, particularly in cases when there is a class imbalance. It is especially useful when a compromise needs to be struck between reducing false and positives as well as false and negatives.

The harmonic mean of recall and precision calculates the F1 score. The definition of it in mathematics is:

F1 Score = 2 * (Recall * Precision) / (Recall + Precision). (4)

An improvement in model performance is demonstrated by an increase in the F1 score from 0 to 1. It functions best when the positive and negative classes are distributed unevenly or when the costs of false positives and false negatives fluctuate considerably.

The F1 score offers a rather full evaluation of a model's performance than metrics that address one of these factors since it accounts for false recall positives and false recall negatives. By combining precision and recall, the F1 score ensures that the model's ability to correctly find positive examples (precision) and capture all positive occurrences (recall) is given equal weight.

Note that the F1 score identifies a threshold that most closely matches the desired result in circumstances where false positives and false negatives have varying effects. For instance, the F1 score assists in choosing a threshold that balances the cost of false negatives against false positives in medical diagnostics when missing positive cases can be extremely crucial.

However, it is vital to remember that the F1 score might not always be the best statistic, particularly if the demands for recall and precision differ. Depending on the application, you might prioritize accuracy or recall; in these situations, other metrics like precision-recall or ROC (receiver operating characteristic) curves may offer more insightful data. The F1 score is a helpful tool for assessing a model's performance when false positives and false negatives must be considered and balanced since it combines accuracy and recall into a single flexible statistic. There are two types of classes in Figure 6, one for cars and one for parking spaces. The figure specified that the F1 value is 1.00 with a 0.637 confidence level for the parking space and the vehicle.

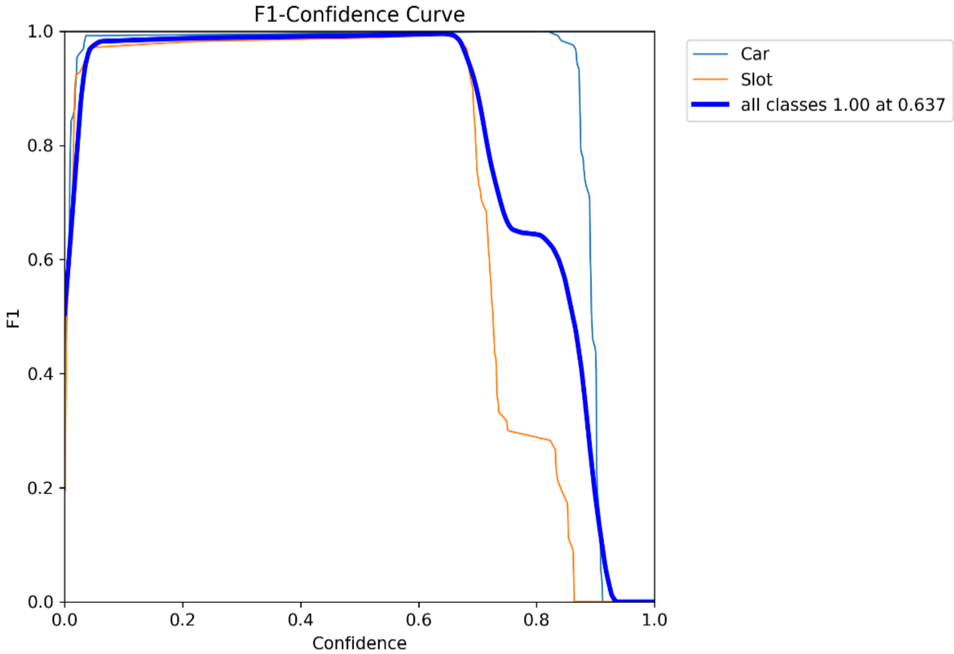

Fig. 6. F1-Confidence curve

4. Results and Discussion

This section provides a thorough summary of performance results displayed by the generated model under various testing circumstances. The assessment focuses primarily on assessing the model's efficiency in terms of detection speed and accuracy, as well as any potential impact of changing the detection approach of edge on its ability to recognize cars and parking spaces. The model used the YOLOv5 algorithm, a strong framework known for its effectiveness in categorizing cars in parking lots for object identification and classification. The evaluation's results offer an understanding of the model's capabilities across multiple media, including pictures and videos.

Other than that, the ability of the model to quickly detect objects is the key component of its competence. The model's practical usefulness is significantly influenced by the speed of detection, a crucial component of real-time applications. Furthermore, a key component of its dependability is detection accuracy. The model's aptitude to be a dependable tool in various disciplines is highlighted by its ability to distinguish items accurately.

The evaluation's examination into how modifications to the edge detection technique may affect how accurately the model detects cars and parking spaces is one prominent area of concentration. As different edge detection techniques may differ in efficacy, this parameter provides insights into the model's adaptability and robustness across diverse contexts and settings.

Correspondingly, the findings of this thorough evaluation suggest that the model had a noteworthy level of effectiveness in cars located in the parking lot. The average detection accuracy was a remarkable 80%, indicating high precision. The model's dynamic performance spectrum was demonstrated by the detection accuracy, which varied between 70 and 98%.

This section clarifies how the constructed model performs under various testing scenarios as a crucial link between theory and practical application. The YOLOv5 algorithm's use and the resulting detection results highlight the model's potential to be useful across a range of domains that depend on quick and accurate object detection. The differences in accuracy and effect of edge detection highlight the model's versatility, which necessitates additional investigation and potential improvement (Figure 7).

Fig. 7. The image shows the performance of YOLOv5 when used to identify cars and slots

Figure 8 displays that 100 epochs were applied to the dataset, which required about 0.055 hours of processing time. The climax of the precision, recall, and accuracy measures was measured after these 100 epochs had passed. The final precision, recall, and accuracy values for all kinds were 0.986, 1, and 0.989, respectively. Particularly noteworthy were the final precision, recall, and accuracy values for the "car" category, which were 0.9995, 1, and 0.995, respectively. The model also performed well in the "parking slot" category, with final metrics for precision, recall, and accuracy of 0.986, 1, and 0.989, respectively.

Other than that, the prototype for an automated parking system made use of YOLO's object-detecting capabilities. The model was validated on five video streams demonstrating various parking circumstances. Correspondingly, these video streams captured instances of cars pulling into and out of parking spaces, and the model demonstrated an impressive level of accuracy in classifying both cars and parking spaces.

The testing procedure included two alternative modes in addition to the standard representation of parking conditions: one enabled motion detection for vehicles entering the parking facility, and the other disabled this function, as displayed in Figure 8. The model's adaptability and efficacy in various contexts are highlighted by this thorough study, which also sheds light on how well it performs in various situations.

```
      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      96/99      4.33G    0.02488    0.03572  0.0006007        124        640: 100% 5/5 [00:01<00:00,  4.37it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:00<00:00,  4.99it/s]
                   all         35        147      0.985          1      0.989      0.603

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      97/99      4.33G    0.02556    0.03767  0.0007313        118        640: 100% 5/5 [00:01<00:00,  4.26it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:00<00:00,  6.07it/s]
                   all         35        147      0.986          1      0.989      0.597

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      98/99      4.33G    0.02475    0.03432  0.0006246        110        640: 100% 5/5 [00:01<00:00,  4.67it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:00<00:00,  4.59it/s]
                   all         35        147      0.986          1      0.989      0.594

      Epoch    GPU_mem   box_loss   obj_loss   cls_loss  Instances       Size
      99/99      4.33G    0.02509    0.03838  0.0005703        127        640: 100% 5/5 [00:01<00:00,  4.30it/s]
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:00<00:00,  5.53it/s]
                   all         35        147      0.986          1      0.989      0.594

100 epochs completed in 0.055 hours.
Optimizer stripped from runs/train/exp/weights/last.pt, 14.4MB
Optimizer stripped from runs/train/exp/weights/best.pt, 14.4MB

Validating runs/train/exp/weights/best.pt...
Fusing layers...
Model summary: 157 layers, 7015519 parameters, 0 gradients, 15.8 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100% 2/2 [00:00<00:00,  3.33it/s]
                   all         35        147      0.986          1      0.989      0.605
                   Car         35         63      0.995          1      0.995      0.719
                  Slot         35         84      0.976          1      0.983       0.49
Results saved to runs/train/exp
```

Fig. 8. The results below show the performance of YOLOv5 with 100 epochs

In Figure 9, the values of Figure 8 are represented by graph plotting. The first three graphs in the first row present the box_loss, obj_loss, and cls_loss values through a curve during the training phase. In the train and box_loss graph, for the highest point, the value of the y-axis is near 0.11, while the value of the x-axis is 100. In the train and obj_loss graph, the highest value of the graph on the y-axis is near 0.07, and on the x-axis, the largest value is 100. In the train and cls_loss graph, the highest value of the graph on the y-axis is near 0.025, and on the x-axis, the largest value is 100. The last two images of the first row illustrate the metrics of precision and recall value. In the matrics and precision graph, the highest point, the value of the y-axis, is near 1.0, and the value of the x-axis is 100. In the matrics and recall graph, the highest point, the value of the y-axis, is near 1.0, and the value of the x-axis is 100.

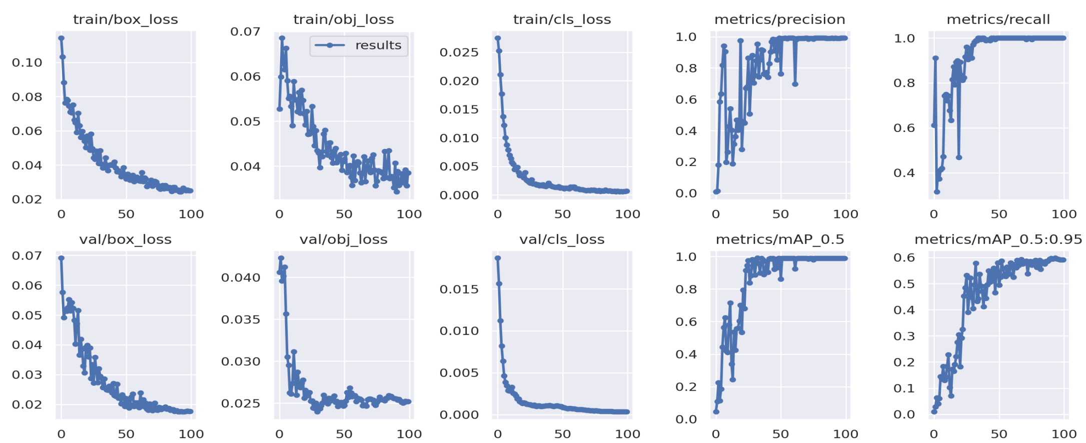

Fig. 9. The total result for presenting for accuracy, precision, and recall in the graph

In the second row, box_loss, obj_loss, and cls_loss values are presented through a curve during the training phase. In the validation and box_loss graph, for the highest point, the value of the y-axis is near 0.07, and the value of the x-axis is 100. In the validation and obj_loss graph, for the highest point, the value of the y-axis is near 0.040, and the value of the x-axis is 100. In the validation and box_loss graph for the highest point, the value of the y-axis is near 0.015, and the value of the x-axis is 100. The last two images of the second row provide the accuracy matrics value. In the matrics and accuracy graph, the highest point, the value of the y-axis, is 1.0, the value of the x-axis is 0, and the accuracy is 0.5. In the last matrics and accuracy graph, the highest point is that the value of the y-axis is 0.6, the value of the x-axis is 0, and the accuracy is 0.5:0.95.

5. Conclusions

The central objectives of this research encompassed a meticulous analysis of the techniques presently employed in the realm of automotive parking detection. It is coupled with developing and evaluating a model with the innate ability to predict parking spots with a notably high precision level. This pursuit entailed the utilization of the YOLOv5 model, an established framework known for its prowess in object detection, specifically tailored to discern automobile objects within parking lots. Crucially, the training process was conducted using an in-house dataset, meticulously curated to encompass standardized items, ensuring the model's optimal learning.

A user-friendly software program was developed with a course of study to make recording coordinates for each parking space much easier. Other than that, the designation of parking spaces made possible by this application served as a solid basis for future investigation. Following the input of the coordinates, a classifier was used to establish the parking lot's occupancy state by recognizing the presence of cars.

The journey of object recognition commenced shortly after the acquisition of images from the video stream, with the loaded images being fed into the model. The model's proficiency in object detection was the first checkpoint. Subsequently, an evaluation was executed to ascertain the status of each parking spot. Notably, in cases where the status of parking spots remained ambiguous, a motion detection technique was seamlessly integrated. This technique, deployed to augment the accuracy of the model, aimed to rectify parking spot statuses that could not be definitively determined. By sensing motion around these uncertain spots, the model reassessed their status after a brief interval, enhancing the model's ability to reflect the most current state of the parking lot.

This methodological approach extends beyond the laboratory setting, exhibiting the potential for real-world applications in environments such as shopping malls, urban centres, and airports—locations characterized by a perpetual flux of vehicles entering and exiting parking lots. In the conventional scenario, drivers often navigate parking areas aimlessly, searching for vacant parking spots, resulting in lost time and compounding traffic congestion. Thus, the model developed in this research could revolutionize this experience, arming drivers with prior knowledge of parking spot availability. It also enables them to decide about entering or seeking alternative spots. Accordingly, this proactive approach to parking would be facilitated by the model's insights into the current status of parking zones.

It is noteworthy that the scope of this work is tailored specifically for parking garages rather than open-air parking areas. Furthermore, its application is limited to a specific corner area without encompassing a comprehensive around-view perspective. Therefore, enhancements in the model's design could potentially involve the incorporation of user notifications, allowing individuals to receive real-time alerts whenever a parking spot is identified, thus optimizing the overall parking experience.

As the research embarks on future horizons, there is a desire to explore more advanced versions of YOLO, aiming to harness a higher degree of accuracy across varying datasets. Nevertheless, the model's potential evolution holds promise in addressing diverse challenges and complexities inherent in various parking scenarios, contributing to enhanced efficiency, reduced congestion, and an improved parking landscape overall.

Acknowledgement
This research was partially funded by the Centre of Excellence for Advanced Computing (ADVCOMP), Universiti Malaysia Perlis.

References
[1] D. Trivedi, Janak, Sarada Devi Mandalapu, and Dhara H. Dave. "Real-time parking slot availability for Bhavnagar, using statistical block matching approach." *World Journal of Engineering* 17, no. 6 (2020): 811-821. https://doi.org/10.1108/WJE-09-2019-0263
[2] Valipour, Sepehr, Mennatullah Siam, Eleni Stroulia, and Martin Jagersand. "Parking-stall vacancy indicator system, based on deep convolutional neural networks." In *2016 IEEE 3rd World Forum on Internet of Things (WF-IoT)*, pp. 655-660. IEEE, 2016. https://doi.org/10.1109/WF-IoT.2016.7845408
[3] Paidi, Vijay, Hasan Fleyeh, and Roger G. Nyberg. "Deep learning-based vehicle occupancy detection in an open parking lot using thermal camera." *IET Intelligent Transport Systems* 14, no. 10 (2020): 1295-1302. https://doi.org/10.1049/iet-its.2019.0468
[4] Mahmud, Rifath, AFM Saifuddin Saif, and Dipta Gomes. "A Comprehensive Study of Real-Time Vacant Parking Space Detection Towards the need of a Robust Model." *AIUB Journal of Science and Engineering (AJSE)* 19, no. 3 (2020): 99-106. https://doi.org/10.53799/ajse.v19i3.80
[5] Amato, Giuseppe, Fabio Carrara, Fabrizio Falchi, Claudio Gennaro, and Claudio Vairo. "Car parking occupancy detection using smart camera networks and deep learning." In *2016 IEEE symposium on computers and communication (ISCC)*, pp. 1212-1217. IEEE, 2016. https://doi.org/10.1109/ISCC.2016.7543901
[6] Acharya, Debaditya, and Kourosh Khoshelham. "Real-time image-based parking occupancy detection and automatic parking slot delineation using deep learning: A tutorial Indoor mapping, modeling and localization View project Real-time image-based parking occupancy detection and automatic parking slot deliniation using deep learning: A tutorial." (2020).
[7] Šćekić, Zoja, Stevan Čakić, Tomo Popović, and Anja Jakovljević. "Image-based parking occupancy detection using deep learning and faster r-cnn." In *2022 26th international conference on information technology (IT)*, pp. 1-5. IEEE, 2022. https://doi.org/10.1109/IT54280.2022.9743533
[8] Nyambal, Julien, and Richard Klein. "Automated parking space detection using convolutional neural networks." In *2017 Pattern Recognition Association of South Africa and Robotics and Mechatronics (PRASA-RobMech)*, pp. 1-6. IEEE, 2017. https://doi.org/10.1109/RoboMech.2017.8261114
[9] Jiang, Shaokang, Haobin Jiang, Shidian Ma, and Zhongxu Jiang. "Detection of parking slots based on mask R-CNN." *Applied Sciences* 10, no. 12 (2020): 4295. https://doi.org/10.3390/app10124295
[10] Ibisch, André, Stefan Stümper, Harald Altinger, Marcel Neuhausen, Marc Tschentscher, Marc Schlipsing, Jan Salinen, and Alois Knoll. "Towards autonomous driving in a parking garage: Vehicle localization and tracking using environment-embedded lidar sensors." In *2013 IEEE intelligent vehicles symposium (IV)*, pp. 829-834. IEEE, 2013. https://doi.org/10.1109/IVS.2013.6629569
[11] Tong, Lihua, Liang Cheng, Manchun Li, Jiechen Wang, and Peijun Du. "Integration of LiDAR data and orthophoto for automatic extraction of parking lot structure." *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing* 7, no. 2 (2013): 503-514. https://doi.org/10.1109/JSTARS.2013.2269193
[12] Yu, Luyang, Haobin Jiang, and Lei Hua. "Anti-congestion route planning scheme based on Dijkstra algorithm for automatic valet parking system." *Applied Sciences* 9, no. 23 (2019): 5016. https://doi.org/10.3390/app9235016
[13] Jeong, S. H., C. G. Choi, J. N. Oh, P. J. Yoon, B. S. Kim, M. Kim, and K. H. Lee. "Low cost design of parallel parking assist system based on an ultrasonic sensor." *International Journal of Automotive Technology* 11 (2010): 409-416. https://doi.org/10.1007/s12239-010-0050-0
[14] Nam, Yunyoung, and Yun-Cheol Nam. "Vehicle classification based on images from visible light and thermal cameras." *EURASIP journal on image and video processing* 2018 (2018): 1-9. https://doi.org/10.1186/s13640-018-0245-2
[15] Irhebhude, Martins E., Mohammad Athar Ali, and Eran A. Edirisinghe. "Pedestrian detection and vehicle type recognition using CENTROG features for nighttime thermal images." In *2015 IEEE International Conference on Intelligent Computer Communication and Processing (ICCP)*, pp. 407-412. IEEE, 2015. https://doi.org/10.1109/ICCP.2015.7312693

[16] Sangnoree, Apiwat, and Kosin Chamnongthai. "Thermal-image processing and statistical analysis for vehicle category in nighttime traffic." *Journal of Visual Communication and Image Representation* 48 (2017): 88-109. https://doi.org/10.1016/j.jvcir.2017.06.006
[17] Amato, Giuseppe, Fabio Carrara, Fabrizio Falchi, Claudio Gennaro, Carlo Meghini, and Claudio Vairo. "Deep learning for decentralized parking lot occupancy detection." *Expert Systems with Applications* 72 (2017): 327-334. https://doi.org/10.1016/j.eswa.2016.10.055
[18] Lin, Jia-Ping, and Min-Te Sun. "A YOLO-based traffic counting system." In *2018 Conference on Technologies and Applications of Artificial Intelligence (TAAI)*, pp. 82-85. IEEE, 2018. https://doi.org/10.1109/TAAI.2018.00027
[19] Hossin, Mohammad, and Md Nasir Sulaiman. "A review on evaluation metrics for data classification evaluations." *International journal of data mining & knowledge management process* 5, no. 2 (2015): 1. https://doi.org/10.5121/ijdkp.2015.5201
[20] Butt, Muhammad Atif, Asad Masood Khattak, Sarmad Shafique, Bashir Hayat, Saima Abid, Ki-Il Kim, Muhammad Waqas Ayub, Ahthasham Sajid, and Awais Adnan. "Convolutional neural network based vehicle classification in adverse illuminous conditions for intelligent transportation systems." *Complexity* 2021, no. 1 (2021): 6644861. https://doi.org/10.1155/2021/6644861
[21] Rafique, Sarmad, Saba Gul, Kaleemullah Jan, and Gul Muhammad Khan. "Optimized real-time parking management framework using deep learning." *Expert Systems with Applications* 220 (2023): 119686. https://doi.org/10.1016/j.eswa.2023.119686
[22] Khan, Gulraiz, Muhammad Ali Farooq, Zeeshan Tariq, and Muhammad Usman Ghani Khan. "Deep-learning based vehicle count and free parking slot detection system." In *2019 22nd International multitopic conference (INMIC)*, pp. 1-7. IEEE, 2019. https://doi.org/10.1109/INMIC48123.2019.9022687
[23] Ling, Xiao, Jie Sheng, Orlando Baiocchi, Xing Liu, and Matthew E. Tolentino. "Identifying parking spaces & detecting occupancy using vision-based IoT devices." In *2017 Global Internet of Things Summit (GIoTS)*, pp. 1-6. IEEE, 2017. https://doi.org/10.1109/GIOTS.2017.8016227
[24] Zinelli, Andrea, Luigi Musto, and Fabio Pizzati. "A deep-learning approach for parking slot detection on surround-view images." In *2019 IEEE intelligent vehicles symposium (IV)*, pp. 683-688. IEEE, 2019. https://doi.org/10.1109/IVS.2019.8813777
[25] Huang, Ching-Chun, Yu-Shu Tai, and Sheng-Jyh Wang. "Vacant parking space detection based on plane-based Bayesian hierarchical framework." *IEEE Transactions on Circuits and Systems for Video Technology* 23, no. 9 (2013): 1598-1610. https://doi.org/10.1109/TCSVT.2013.2254961
[26] Ichihashi, Hidetomo, Akira Notsu, Katsuhiro Honda, Tatsuya Katada, and Makoto Fujiyoshi. "Vacant parking space detector for outdoor parking lot by using surveillance camera and FCM classifier." In *2009 IEEE international conference on fuzzy systems*, pp. 127-134. IEEE, 2009. https://doi.org/10.1109/FUZZY.2009.5277099
[27] Cheming, Hartinee, and Dawood Abdulmalek Yahya Al-Hidabi. "An Evaluation of Secondary School Thai Language Textbooks in Pattani, Thailand: An Islamic Perspectives." *International Journal of Advanced Research in Future Ready Learning and Education* 34, no. 1 (2024): 113-123. https://doi.org/10.37934/frle.34.1.113123
[28] Hisham, Sri Dewi, Shaza Eva Mohamad, Mohd Ibrahim Shapiai, Koji Iwamoto, Aimi Alina Hussin, Norhayati Abdullah, and Fazrena Nadia Md Akhir. "Comparison of Conventional CNN Sequential API and Functional API for Microalgae Identification." *Journal of Advanced Research in Micro and Nano Engineering* 17, no. 1 (2024): 96-104. https://doi.org/10.37934/armne.17.1.96104
[29] Osman, Manal, Suhaimi B. Hassan, and Khamaruzaman B. Wan Yusof. "Effect of combination factors of operating pressure, nozzle diameter and riser height on sprinkler irrigation uniformity." *Applied Mechanics and Materials* 695 (2015): 380-383. https://doi.org/10.4028/www.scientific.net/AMM.695.380