Parking Lot Vacancy Notification System

Team I-dle
Kim Seheon, Kim Hyeongjin, Noh Hayeon, Park Inchan

October 3, 2024

Contents

Abstract

Parking congestion is a common issue in public spaces, particularly in school environments, where finding an available parking spot is often time-consuming. This project proposes a real-time parking management system that utilizes the YOLOv5 object detection model to identify vacant parking spaces and assess vehicle sizes. The system not only detects empty spots but also compares the size of a vehicle to the available space, providing warnings if the vehicle is too large. The proposed solution aims to optimize parking space usage, reduce congestion, and improve user convenience through an intuitive app interface. By integrating real-time AI-powered detection with vehicle size analysis, this system addresses both parking efficiency and safety concerns.

Keywords: parking management system, YOLOv5, object detection, AI-powered parking solution

1 Introduction

Six out of 10 Korean drivers are stressed by parking problems on a daily basis[2], and nine out of 10 have wasted time even after reaching their destination due to parking problems[1]. This is a serious situation, and it is urgently needed to be addressed, as parking stress is becoming a social problem, with fistfights becoming a common occurrence. In particular, parking has become an important issue in public spaces such as universities, where students and faculty often spend unnecessary time looking for a parking space, causing traffic congestion, unnecessary energy consumption, and environmental pollution.

Parking lot expansion essentially solves the problem of parking shortages, but it's a difficult solution to use in schools, where most spaces are already compressed[3]. In response, parking apps that can check parking spaces in real time have emerged as a solution[1], but there is no service targeting Sungkyunkwan University. In addition, the existing parking system mainly detects only the availability of parking spaces and does not consider the size of the vehicle. It can lead to a decrease in user convenience if parking spaces are provided uniformly despite the fact that parking spaces vary depending on the size of the vehicle, and even worse, if there are not enough parking spaces available, large vehicles may try to park in small spaces, causing traffic congestion or increasing the risk of accidents. Therefore, there is a need for a smart parking management system customized for Sungkyunkwan University that considers vehicle size.

This project proposes a system that utilizes the YOLOv5 object detection model to simultaneously recognize empty parking spaces and vehicle size. The system not only detects empty spaces in real time, but also compares the size of the vehicle to the parking space to determine whether it can be parked, and provides a warning notification to the user if the vehicle size is larger than the parking space. This maximizes the space utilization of the parking lot and prevents accidents caused by incorrect parking of large vehicles.

In this proposal, we will elaborate on the goals and motivation for this project, then review the existing research on parking management and the limitations of the systems. We will also discuss how the proposed solution can address these issues, and detail the system structure, equipment, data collection and model training process, user interface and functionality, and implementation of the system. Finally, we will discuss future plans and role division.

2 Motivation / Objective

The problem of parking space shortage is getting worse. The number of vehicles is constantly increasing, while parking spaces are limited and often not used efficiently. Traditional parking management systems are limited to detecting empty parking spaces and do not take into account the size of the vehicle and the suitability of the space. Due to these limitations, when a large vehicle tries to park in a small parking space, it not only wastes parking space but also increases the risk of accidents.

To address these issues, we propose the following specific project goals.

Project Goals

Real-time parking space detection: Monitor parking spaces in real-time using the YOLOv5 model and detect vacancies.

Recognize vehicle size: Analyze the size of the vehicle, recommend spaces where it can park, and provide warning notifications if the vehicle is larger than the space.

Provide a user-friendly interface: Provide users with real-time availability information via a mobile application, with an intuitive UI that makes it easy to see alerts about parking availability and vehicle size conformity.

Optimize space utilization: Solve the problem of large vehicles parking in unsuitable spaces and maximize the space efficiency of the parking lot.

Reduce parking stress: Reduce the inconvenience and stress of parking by enabling users to quickly find a vacant space and be offered an appropriate parking spot.

3 Background / Related Work

1) Smart Parking Management Using UAVs[6] Another study proposed a smart parking management system using Unmanned Aerial Vehicles (UAVs) to monitor parking lots from above. The system provides real-time, aerial monitoring of parking availability. However, the high cost of UAV deployment and the challenge of providing detailed, close-up data about individual parking spaces remain significant limitations. Our project improves upon this by using a camera-based YOLOv5 system, which is more cost-effective and capable of delivering precise real-time information about parking availability.

2) Parking Space Detection Using YOLOv5[5] This research focuses on utilizing the YOLOv5 model for detecting vacant parking spaces and vehicles in real time. The study highlights the improved accuracy and faster object detection performance of the YOLOv5 architecture. It is highly efficient for real-time parking space analysis, helping optimize parking space usage. However, one limitation of this system is that it does not account for vehicle size, which can lead to inefficient use of space when large vehicles park in small spots. Our project builds upon this by adding a feature to detect vehicle size and match it with the appropriate parking space.

3) Multi-Camera Parking Space Management System[7] This research explored the use of a multi-camera system for managing parking spaces by tracking vehicles across several cameras and determining parking occupancy. While effective, the cost of installing multiple cameras is prohibitive, and the system lacks the ability to assess vehicle size against parking space dimensions. Our project advances this approach by incorporating YOLOv5's capability to detect vehicle size and match it with the appropriate parking space, offering a more comprehensive parking management solution.

4 Problem Statement / Proposed Solution

4.1 Overall System Structure

The proposed system comprises four key components: cameras, an AI model, a database, and a mobile application. Initially, multiple cameras capture real-time footage of the parking lot. This data undergoes a pre-processing phase before being provided as input to the AI model. The AI model analyzes the captured footage, labeling occupied parking spaces, available spots, and large vehicles such as trucks. The output from the AI model is subsequently transmitted to the database, where it is further processed before being sent to the user's mobile device, allowing the results to be displayed.

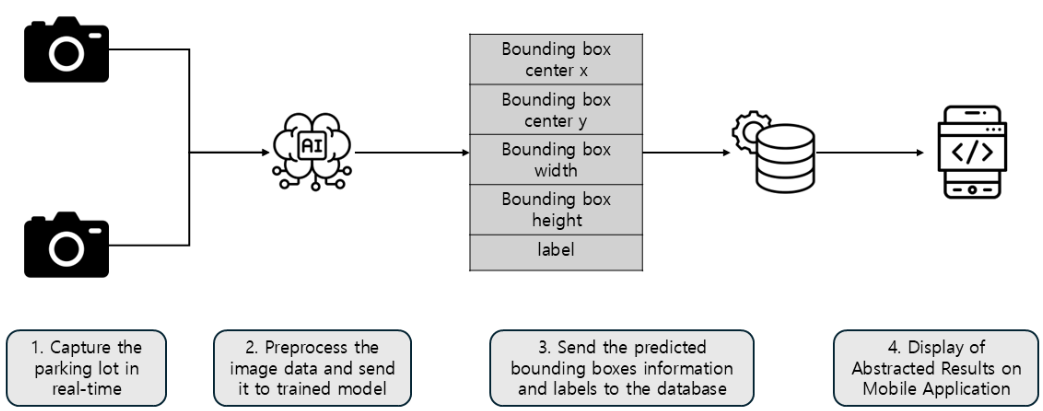

Figure 1: System Architecture

A. Real-Time Capture with Cameras

To cover a large parking area, multiple cameras (two or three) must be installed on the rooftop of the adjacent building. These cameras can either transmit live video feeds or, to reduce operational costs, capture still images at regular intervals. The latter approach enables time-based analysis of parking space utilization, providing a more cost-effective solution. Due to the use of multiple cameras, the system must remove overlapping areas and filter out noise such as excessive lighting, which will be addressed during the pre-processing phase. The processed data is then used as input for the AI model.

B. Object Detection and Labeling

The AI model employed in this system is YOLO (You Only Look Once), which processes the real-time data received from the cameras to label parking spaces. We will use a model pre-trained on the COCO dataset and fine-tune it with the parking space dataset. The model identifies whether each parking space is vacant or occupied and assigns a unique label to each spot. For labeling, the AI's prediction results are mapped and assigned an ID by comparing them with the coordinates of the actual parking lot. These results are then transmitted to the server's database for use in subsequent processes. Additionally, the AI model detects larger vehicles, such as trucks, and uses this information to inform users about the reduced availability of adjacent parking spaces due to size constraints.

C. Transmission to Database The AI-generated predictions are restructured for compatibility with the database, which continuously receives, updates, and stores the incoming data. The database also forwards this information to the mobile application. To minimize operational costs, an event handler is utilized, ensuring data is only sent to the mobile app when necessary. The system also accounts for the time required to detect changes in the parking lot and relay this information to the user.

D. Display of Abstracted Results on Mobile Application The final stage involves the mobile application receiving data from the database, which is then displayed to the user in an abstract format. The overall layout of the parking lot is represented visually, with available and occupied spaces distinguished by color-coded indicators for ease of recognition.

4.2 Camera Placement and Parking Lot Coverage

Since a single camera cannot cover the entire parking lot, multiple cameras (two or three) will be employed to ensure complete surveillance of the parking area.

Figure 2: Parking lot captured from the rooftop of an adjacent building

4.3 AI Model Implement

4.3.1 Object Detection

Object detection is a fundamental task in computer vision that combines two important functions, object classification and object localization. Unlike image classification, which simply identifies the presence of an object in an image, object detection goes a step further by also pinpointing its location within the image. This is achieved by drawing bounding boxes around detected objects and assigning them to specific classes.

Object detection models can be divided into two-stage detectors and single-stage detectors. In two-stage detection, the process is split into region proposal stage and object detection stage

4.3.2 YOLO

The YOLO (You Only Look Once) model is one of the most widely used object detection algorithms, renowned for its real-time detection capabilities and efficiency. Unlike traditional object detection methods that often processed with two-stage model, YOLO simplifies the process by making object detection as a single regression problem. This allows YOLO to detect objects and predict their bounding boxes in one forward pass through the neural network, making it significantly faster and allowing real-time process than other models like R-CNN or Faster R-CNN.

YOLO needs input of an image with RGB channels and makes an output tensor of (S, S, B*(5+C)) size where S is size of the grid, B is the number of bounding boxes for each grid cell, 5 is the number of bounding box's center x, y, width, height and confidence score, C is the number of classes. With the output, use NMS(Non-Maximum Suppression) algorithm to select the most appropriate bounding box.

4.3.3 Dataset

Our project will use PKLot dataset which contains images of parking lots extracted from surveilance camera frames[4]. with annotations. The dataset has 12416 images with size of 1280x720 which are splited into 8691 train images, 2483 valid images and 1242 test images. Parking spaces are labeled as occupied or empty and the images are captured on sunny, cloudy, rainy days. 711,856 annotations exist in the images which is 57.3 annotations per image and label occupied has 335,735 annotations and label empty has 376,121 images.

4.3.4 AI system flow

1. Finetuning the pre-trained model
   The initial step involves selecting a pre-trained YOLOv5 model that has already been trained on the COCO dataset. The pre-trained model will then be fine-tuned on the PKLot dataset. Finetuning is essential to adapt the general object detection model to the specific task of parking space detection, where the goal is to distinguish between occupied and empty spaces in parking lots. The model will learn to recognize vehicles and open spaces by adjusting its weights according to the specific features of the parking lot images and its output would be made to classify only two categories.

2. Predict the spaces occupied or empty
   Once the model is trained, it will be used to predict whether each parking space in the input image of SKKU parking lot is occupied (vehicle detected) or empty (no vehicle detected). This step involves the model scanning the image, dividing it into grid cells, and making predictions about whether each cell contains a vehicle or an open space. Bounding boxes will be drawn around detected vehicles, and each prediction will include a confidence score indicating how certain the model is about the detection..

3. Match the space with parking lot position number
   After identifying which parking spaces are occupied or empty, the system will map these predictions back to the specific parking spots in the parking lot. This is done by comparing the

coordinates of the detected bounding boxes with a predefined layout of the parking lot. Each parking space has a corresponding ID number, allowing the system to provide users with precise information about which specific parking spots are available or occupied.

4.3.5 Limitations

1. Small Object Detection

   If vehicles are too far or hidden by other objects, it would be hard to detect by the YOLO model. This problem might result the model miss some vehicles and make wrong predictions

2. Relatively low accuracy

   YOLO model is fast and can process images in real-time but it results low accuracy than other object detection models. It would be important to adjust the hyperparameters to make accuracy high enough.

4.4 User Interface

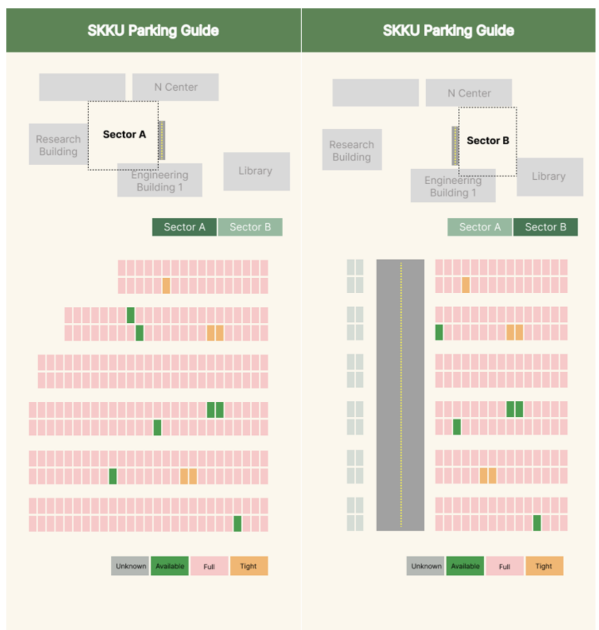

Figure 3: User Interface

4.5 AI and App Stacks

4.5.1 AI Stack

The AI stack focuses on the development and deployment of the machine learning model for detecting parking spaces. Here are the components involved:

Model Training & Development

- **PyTorch**: A Deep learning framework used for building and training the YOLO model or other neural networks. PyTorch provides extensive libraries for designing the object detection model and performing backpropagation during training, allowing customization and fine-tuning of model architecture.

- **OpenCV**: OpenCV is an open-source computer vision library used for image pre-processing tasks, such as resizing, normalization, and augmentation, before feeding parking lot images into the YOLO model. It can also be utilized to process images after detection, such as highlighting the detected bounding boxes or overlaying information about parking spots.

- **Google Colab**: Environments for experimenting, training, and fine-tuning models. Google Colab offers free GPU support, useful for training deep learning models efficiently.

APIs & Integration

- **FastAPI**: Once the model has been trained, FastAPI can be used to deploy the model as a REST API. The REST API will expose endpoints that the Android app can call to upload parking lot images, receive predictions on available parking spaces, and retrieve processed data such as bounding box sizes and center spot.

4.5.2 App Stack

- **Android Studio**: It is a comprehensive app development tool that will be used in conjunction with the Android SDK for building the app's user interface and core logic. Its integrated environment allows for seamless testing and debugging on various Android devices, making it ideal for the project's development needs.

- **Firebase**: Firebase will be utilized to store parking lot data in real-time and synchronize it efficiently with the client. Its cloud-based database system ensures reliable scalability, enabling seamless management and quick access to structured parking data across multiple devices.

4.6 Additional Functions

A system that solely identifies available or occupied parking spots may not fully meet user expectations, nor provide a distinct advantage over existing solutions. In practice, even if a spot is available, users may hesitate to park due to space limitations, particularly when adjacent to large vehicles. To address this issue, the AI model will additionally identify large vehicles such as trucks and notify users if nearby parking spots are constrained. This information will be visually represented in the mobile application, allowing users to make informed decisions and avoid narrow parking spots when possible. This added functionality enhances user experience, providing a more comprehensive and differentiated parking solution.

5 Planning in detail

5.1 Weekly Progression

- Topic proposal: week 3∼ 4
- Data preprocessing: week 5∼ 6
- Model Train: week 7∼ 8
- Model Optimization and Evalution: week 8∼ 9
- UI/UX Design: week 5
- App-frontend: week 6∼ 7
- App-backend: week 8∼ 9
- Integration of model and app: week 10
- Testing, Revision: week 11∼ 12

5.2 Roles & Responsibilities

Name	Role
Kim Seheon	Database, Application Design
Kim Hyeongjin	Application Design, Architecture Design
Noh Hayeon	Vision processing, Communication
Park Inchan	Vision processing, Communication

References

[1] 김건호. (2021). “차 댈 곳 없어 고생 마세요”... 빈자리 알려주는 ‘주차앱’ 뜬다 [심층기획 - ’주차전쟁’ 해법 없나]. 세계일보. https://www.segye.com/newsView/20210111513455.

[2] 김대섭. (2016). 국내 운전자 10명 중 6명 “주차문제로 스트레스”. 아시아경제. https://v.daum.net/v/20160422120713894

[3] 채성숙. (2020). [트렌드모니터] 극심한 ‘주차난’, 근본적인 해결책은 주차공간 확보지만 “글쎄... MAD-TIMES. https://www.madtimes.org/news/articleView.html?idxno=6306

[4] Almeida, P., Oliveira, L. S., Silva Jr, E., Britto Jr, A., Koerich, A., PKLot – A robust dataset for parking lot classification, Expert Systems with Applications, 42(11):4937-4949, 2015.

[5] D.-L. Nguyen et al. (2023). YOLO5PKLot: A Parking Lot Detection Network Based on Improved YOLOv5 for Smart Parking Management System. Springer. https://doi.org/10.1007/978-981-99-4914-4_8

[6] Li, Xin & Chuah, Mooi Choo & Bhattacharya, Subhrajit. (2017). UAV assisted smart parking solution. 1006-1013. 10.1109/ICUAS.2017.7991353.

[7] R. Martín Nieto, Á. García-Martín, A. G. Hauptmann and J. M. Martínez. (2019). Automatic Vacant Parking Places Management System Using Multicamera Vehicle Detection” in IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 3, pp. 1069-1080. https://doi.org/10.1109/TITS.2018.2838128