UDC 004.032.26

THE SERVER PART OF THE ONLINE PARKING BOOKING SYSTEM

A. KHODOSEVICH, R. BOGUSCH
Polotsk State University, Belarus

The article deals with the implementation of the server part of the parking space system using a neural network. The principles and possible application of recognition of parking spaces and cars in the image are given.

Road transport is an integral part of modern society. With the growing number of cars in each city, the problem of parking is becoming more and more urgent every day. This issue is even more acute in shopping and entertainment complexes, restaurants, hotels, airports, supermarkets and nightclubs, where the competent organization of parking lots is directly related to their life support and profitability.

As long as the number of cars did not exceed a certain threshold, they could be parked at the roadsides and in other places near the desired object. But, first of all, it was not always possible to do this by observing the traffic rules. And secondly, leaving the car unattended is always associated with the risk of its theft, burglary or damage to property. This task in the city is solved by the organization of paid parking near infrastructure facilities [1].

In cities, there are a huge number of parking spaces, some of them are paid and to book a place, you need to contact the manager using a phone call. In addition, information technologies are actively developing, more and more devices are accessing the Internet, and more and more applications are moving online. Information technology has become an integral part of every person's life. Thus, it is relevant to use software to track available parking spaces based on neural networks and then book these parking spaces.

One of the most promising areas of artificial intelligence is neural networks. Now they are actively used in business, used in the field of security, entertainment and other areas. Artificial neural networks are built on the principle of biological ones, of course, with a number of assumptions, they operate a huge number of simple processes with many connections. Like the human brain, these networks are capable of learning [2].

The main task of the parking space management system is to accurately determine the available parking spaces. The first step is to recognize all possible parking spaces in the frame. Before you search for unoccupied spaces, you need to determine how the parking space will be recognized. The second step is to recognize all the cars in the frame. The third step is to determine which places are occupied by cars and which are not. To do this, you need to combine the results of the first two steps. Next, the program should display when the parking space is free or occupied. Each of these stages can be completed in different ways using different technologies. There is no single right or wrong solution to these problems, different approaches will have their advantages and disadvantages. Let's look at each step in more detail.

The source data will be an image from a video surveillance camera. This image should be scanned and get a list of places where you can park. To do this, consider several solutions.

The first option is to get an image from the video camera and select all parking spaces manually, instead of automatic recognition. But in this case, if there is a movement/displacement of the camera or use the system in another parking lot, you will have to do the whole procedure again.

The second option is to find an indicator in the image (e.g. a parking meter) and assume that there is a parking space next to each indicator. However, there are some difficulties here, since not every parking space has the ability to install an indicator and this is more expensive. Also, the location of the indicator does not explicitly indicate where the parking space is located and how it is located, but only allows you to make a guess.

The next option is to create an object recognition model that looks for the markings of a parking space located on the road. But this approach has disadvantages, such as the need to apply the marking itself on the road, which may eventually become unusable, and other lines or markers may be present in the parking lot, which may lead to erroneous recognition of the parking space.

The last option is to determine the parking space where cars have been parked for a long time. It is possible to guess where the parking spaces are if you recognize and identify cars that do not move between frames. But even in this option there are disadvantages, since in order to determine the parking space, you need to fill the parking lot with cars. This method is more suitable for determining parking spaces along roads or streets.

As a result, the first option was chosen, because it is more similar to the requirements of the system implementation.

To determine whether a parking space is occupied or not, you need to implement car recognition. On the frame, machine recognition will be a classic object recognition task. There are many machine learning-based approaches that could be used for speech recognition:

– Train the detector based on the Histogram of Oriented Gradients (HOG) and analyze the entire image to identify all the machines. This approach, which does not use deep learning, works relatively quickly, but the quality of finding machines that are located differently does not meet the requirements of the system [3].

– Train the detector based on Convolutional Neural Network (CNN) and analyze the entire image to identify all the machines. This approach works accurately, but not as efficiently, since it requires multiple image scans using CNN to find all the machines. With this method, it will be possible to find machines that are located differently, but this will require much more training data than for a HOG detector [4].

– Use a deep learning approach-Mask R-CNN, Faster-CNN, or YOLO, which combines the accuracy of CNN and a set of technical properties that significantly increase the speed of recognition. Such models will run relatively fast (on the GPU) if we have a lot of data to train the model [5].

In general, it is necessary to choose the simplest solution that will work according to the expected result and will require the least amount of training data. It doesn't have to be the newest and fastest algorithm. Having correlated all the requirements and costs, it follows that the system must use the Mask R-CNN.

The R-CNN Mask architecture is designed in such a way that it recognizes objects in the entire image, effectively wasting resources, and does not use the sliding window approach. In other words, it works quite fast [6]. With a modern GPU, it is possible to recognize objects in high-resolution video. For this system, this will be enough.

In addition, Mask R-CNN provides a lot of information about each recognized object. Most recognition algorithms only return a bounding box for each object. However, Mask R-CNN will not only give us the location of each object, but also its contour (mask).

To train the Mask R-CNN, you need many images of objects that should be recognized. One way is to go outside, take a picture of the cars and label them in the photos, which would take some time. At the moment, there are already several publicly available datasets with images of cars, since cars are one of the most common objects. One of them is the popular Common Objects In Context (COCO) dataset, which has images annotated with object masks. This dataset contains more than 12,000 images with already marked-up machines [7]. Next, you need to train the model using the COCO dataset. At the moment, there is already a trained model, as the mentioned dataset is popular. Therefore, instead of training your own model, you can take a ready-made one that can already recognize cars. For this system, you can use the model from Matterport. In the future, if the model does not match the quality of object recognition, it is possible to train your model. For artificial neural networks, learning refers to the process of configuring the network architecture (the structure of connections between neurons) and the weights of synaptic connections (the coefficients that affect the signals) to effectively solve the task. Training of a neural network is carried out on a certain sample. In the course of training, the network begins to perform tasks better and respond to the assigned commands.

As a result, the Mask R-CNN model returns 4 values for each recognized object:

1. Type of detected object (integer). The pre-trained COCO model can recognize 80 different frequently encountered objects like cars and trucks.
2. The degree of confidence in the recognition results. The higher the number, the more confident the model is that the object is recognized correctly.
3. Bounding box for the object in the form of X Y-coordinates of pixels in the image
4. "Mask", which shows which pixels inside the bounding box are part of the object. You can use the mask. data to find the outline of an object.

As a result, there are pixel coordinates of each machine. Figure 1 shows object recognition from a video camera.

Fig. 1. – Object recognition from an image

After getting the pixel coordinates, it is possible to determine whether the parking space is occupied by a car or not. To do this, each parking space is analyzed in turn and it is determined whether an object of the "car" type is located in this area or not. If the object of the "car" type is located inside the area of the parking space, then it is occupied. If the object of the "car" type is not located inside the parking space area, then the space is free and it can be reserved.

The user makes a choice of parking on the server part of the parking space system receives an image from a video camera that is located in the parking lot. The image is processed by a neural network and determines the location of parking spaces. Also, the system analyzes all parking spaces to determine which of these spaces are occupied and which are free. This data is written to the database.

At this time, the client part is updated and displays the current data. Figure 2 shows the form for selecting a parking space and booking this parking space.

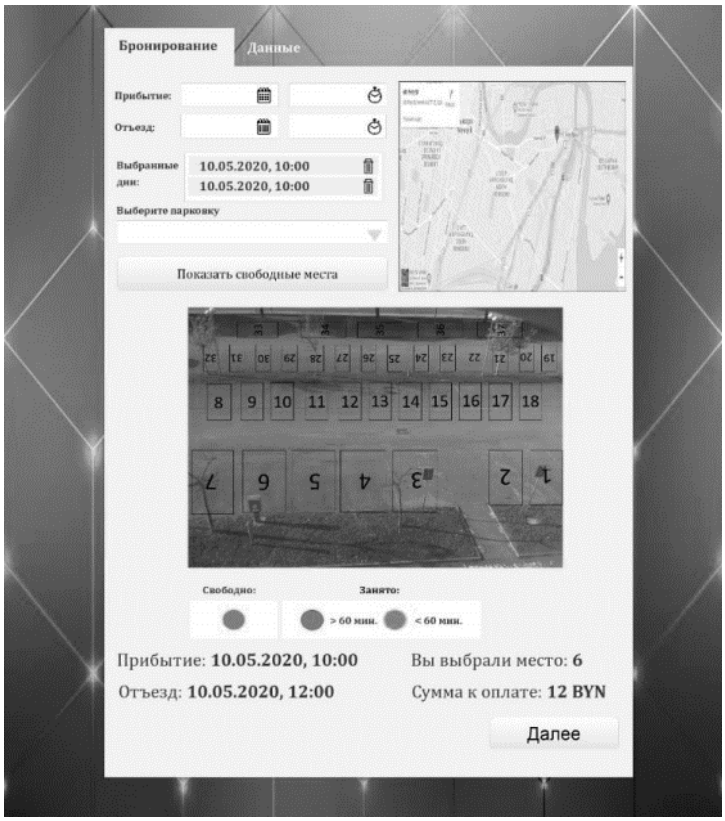

Fig. 2. – Parking space selection form

REFERENCES

1. Проектирование магазинов и торговых центров / Канаян К. А., Канаян Р. В. – Москва, 2015 – 68 с.
2. Нейронные сети: применение сегодня и перспективы развития / Фаустова К.И. – Москва, 2017 – 83 с.
3. Selection of Histograms of Oriented Gradients Features for Pedestrian Detection / Takuya KobayashiAkinori HidakaTakio Kurita – Tsukuba, 2004 – 308 p.
4. Grokking Deep Learning / Andrew Trask – New York, 2019 – 50 p.
5. Mask R-CNN / Kaiming H., Gkioxari G. H., Piotr Dollar, Girshick R. M. - The Hague, 2017 – 41 p.
6. Visualizing and understanding convolutional neural networks / M. D. Zeiler and R. Fergus – Berlin, 2014 – 62 p.
7. Microsoft Common Objects in Context / Michael Maire, Serge Belongie // Microsoft COCO [Electronic resource]. – Washington, 2014. – Mode of access: https://paperswithcode.com/dataset/coco. – Date of access: 10.04.2021.