

AdaBoost for Parking Lot Occupation Detection

Radovan Fusek¹, Karel Mozdřen¹, Milan Šurkala¹ and Eduard Sojka¹

Abstract Parking lot occupation detection using vision systems is a very important task. Many systems use different sensors and their combinations to find out whether the parking lot or space is occupied or not. Using CCTV systems makes it possible to monitor great areas without a need of many sensors. In this paper, we present a method that uses the boosting algorithm for car detection on particular parking spaces and shifting the image to obtain a probability function of car appearance. Using the model of parking lot, we achieve occupancy of each parking space. We also experimented with the detector that is based on the histogram of oriented gradients (HOG) with a support vector machine (SVM). Nevertheless, we found some drawbacks of this detector that we describe in experiments. On the grounds of these drawbacks, we decided to use the AdaBoost based detector.

1 Introduction

Traffic information systems rely on multiple information sources. It is necessary to acquire as much data as possible, but we also require high reliability of this data. Parking lot occupation information systems are useful for intelligent car assistants for on-line planning and finding of spaces for parking.

In some parking lot occupation detection systems, many different sensors are used and it considerably increases the price of whole system. Recently, these sensors are replaced with CCTV systems in many fields. The motivation for this is that one camera placed on parking lot can monitor all parking spaces at once.

Technical University of Ostrava, FEECS, Department of Computer Science,
17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
{radovan.fusek, karel.mozdren, milan.surkala, eduard.sojka}@vsb.cz

The problem of parking lot detection system is to find out occupation of parking spaces. The systems that are available can be divided into two categories, car based systems and parking space based systems. The car based systems are based on finding the cars in the images. The parking space based systems depend on their state with respect to their particular area, which is often combined with the background subtraction models.

In our approach, we work with the images from a camera and we use the AdaBoost based algorithm to find cars in the specific parking space. If it is found, then the parking space is marked as occupied. Otherwise, it is marked as free. Boosting algorithms are quite fast, but there might be many parts of the image that can be similar to cars. We use only a part of car image for detection and this way we try to minimize the error of false detection. Therefore, our method is a combination of car and space driven methods.

We also shift the detection area in the image to compute probability function. The function is computed as a sum of successful car detections. According to this and a preset threshold, we mark the parking space as occupied or not. This makes our approach less false detection prone.

In this paper, we describe our approach in detail. We provide the readers with information on the boosting algorithms and the problem of classification. Then we show the results of our approach and discuss them in the conclusion.

2 Related Work

Many methods for object detection and recognition using machine learning methods were developed recently [8, 11, 3]. The object detection systems can be divided into several classes: feature based, model based, background subtraction based, etc. Related work based on car detection can be also divided into the two main classes [4]: explicit and implicit.

The explicit models [10, 7, 5] use generic models of cars. It is represented as an 2D or 3D model of the car. The models are also often decomposed to sub models and are detected separately according to their features. The combination of these features is then used for the detection of whole car. The most used features are based on rectangular car borders, windshields, intensity of color, or intensity of shades. The car is detected if enough evidence of car presence is found.

Implicit modeling is based on appearance [3, 1, 6]. It uses sample images to describe the model of the car. Classifiers or statistical tools are used in this modeling for classification of the input samples according to the extracted features. For example, the detection methods that are based on the support vector machine (SVM) were presented in [9, 11, 8]. The algorithms for detection are based on finding (matching) the car models in the whole input image. Some methods are focused on contextual-knowledge [14, 7]. In this case, the cars are detected only on roads or parking lots.

The detectors that are based on the histogram of oriented gradients (HOG) with SVM are also capable of this detection task [2]. Nevertheless, this detector had a high number of false positive detections in noisy images. That is the reason why we use the algorithm that are based on the Haar-like features with AdaBoost. We show in our experiments that the proposed detector achieves better results in noise images than the detector that is based on HOG with SVM.

3 Boosting, AdaBoost

Our training data obtained from the parking space can be used with many classifiers. We use the algorithm proposed by Viola and Jones [12, 13] due to its success in many detection tasks and due to the reasons that was described above. Viola and Jones used Haar-like features, integral images, AdaBoost and cascade structure in their framework.

Haar-like features consist of white and black regions and the features are defined as difference of the sum of pixels between the regions. The response to the Haar-like features is computed very quickly using the image representation called integral images. The features can be used as weak classifiers. Generally, many variations of these features can be formulated.

AdaBoost is used to select the most useful and relevant classifiers (features). The main idea of AdaBoost algorithm is to create a strong classifier as a linear combination of weighted simple weak classifiers. Single Haar-like feature can represent a simple weak classifier. The input for AdaBoost is a training set $S = (x_1, y_1), \dots, (x_m, y_m)$, where x_i represent sample (in our case cars) and y_i represent possible evaluation (positive samples are marked as 1, negative samples has 0). The algorithm works in an iterative way and it is changing the importance of data in each its step. Finally, AdaBoost choose effecient features ("weak" classifiers) according to the lowest error and it creates "strong" classifier using a linear combination of weak classifiers.

Viola and Jones object detection framework uses cascade structures to achieve a better classifier speed. The cascade consists of several stages. The stages consist of several simpler classifiers. The main idea of cascade is the rejection of negative candidates in the early stages of cascade. The candidate that successfully passes all phases of cascade is marked as a positive detection (in our case as the car).

4 Our Approach - Proposed Method

Our approach consists of several steps. These steps are shown in the diagram in Figure 1. In the first step, we obtain the image from the camera. The

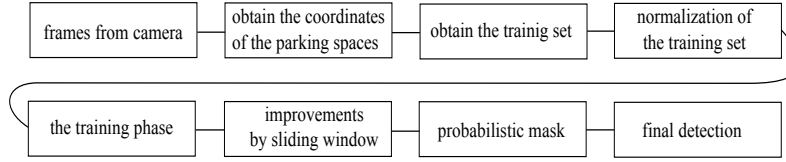


Fig. 1 Diagram containing the basic steps of our approach.

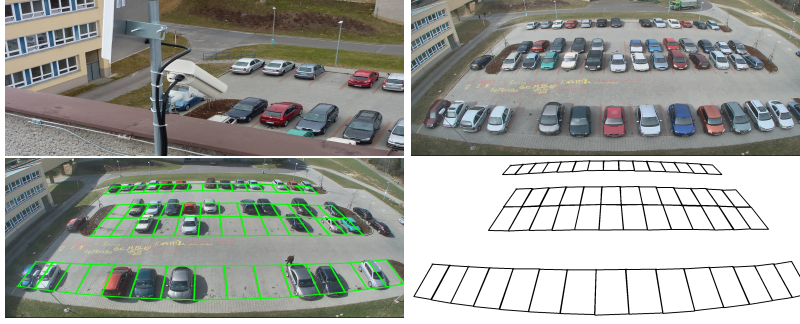


Fig. 2 Diagram containing the basic steps of our approach.

location of our camera is on the roof of the near building that is above the parking lot. Sample images obtained from this camera are shown in Figure 2. In the second step, we get the coordinates of the parking spaces. Each parking space is defined by four coordinates and we manually draw a map of parking spaces. The map of parking spaces is shown in Figure 2. Finally, the coordinates are obtained for each parking space from this map.

After obtaining the coordinates, we can process images from each parking space separately. These images are normalized for the training phase. The normalization includes an important preprocessing steps: a perspective transformation, cut off the background and size normalization. We use a normalized images as an input to the training phase. After the training phase, we obtain our classifier. We use this classifier in the detection phase. In the next section, we suggest the following steps to improve detection in our application of parking space.

4.1 Improvements of Detection Rate

In order to achieve higher detection rate, we propose several techniques that should increase rate of correct car detection and concurrently decrease false positive rate. The first technique is called *window shift*. We search for car not only in the window located in a position of parking space, but we also slightly shift this window in 8 additional directions (right, left, top, bottom,

right-top, left-top, right-bottom and left-bottom). The detection algorithm is carried out in all these positions. The number of car detections in one area (9 windows) tells us about the probability of occupancy. If the number is high and it is close to 9, we can mark this space as an occupied one. On the other hand, if the number is close to zero, there is a high probability that the detection was only accidental. In order to mark these position as correctly as possible, we use a probabilistic function which will be presented later. It is obvious that this approach reduces the situations of accidental detections of cars in empty spaces and detections of empty spaces in occupied spaces.

The second technique is called the *reduction of window*. Because of shifting the window, higher probability of false detection in empty spaces can emerge (despite that it mostly reduces this problem). It is caused by interference of neighboring car to detected area. Therefore, we slightly lower the size of searching window by 5 – 10%. Because of that, the shifted windows do not cover so much the neighboring parking spaces and it reduces the false positive rate. Concurrently, it still preserves the advantage of shifting (probability of detection of wrong parked cars and higher probability of correct result because of the higher number of detections).

The third technique is optional and can be eliminated by better marking of parking spaces. The problem lies in strong perspective distortion of parking spaces near the edge of the image. Therefore, we should not mark the parking spaces in the ground level because a large portion of the car could be depicted in a neighboring parking space or even out of the parking lot. It is more appropriate to mark the parking space approximately in the height of headlamps. On the other hand, it is more simple to mark the spaces in the ground level and adjust the searching window by our third technique of automatic *perspective correction*. This correction should increase the car detection rate in the edges of frame but is also could suffer from the problem of detection of boundary stones and shadows as cars. Therefore, the detector should be learned for such negative images, otherwise, many of empty spaces in the edges could be detected as cars. Our classifier was not trained for such cases yet, so we do not use this improvement in our tests. Preliminary tests showed better results in moderately distorted spaces but higher false positive rates were achieved in extreme edges and this should be addressed in future work.

As mentioned before, the output from the shifting improvement gives us a probabilistic model that increases the reliability of parking space occupancy detection. Each time car is detected in shifted area, a Gaussian function is cumulated in probability image. Once it is done for all parking spaces, we get the probability that gives us information about presence of the car in each part of the image (Fig. 3). Each parking space is then processed separately and a threshold is set to distinguish whether the parking space is occupied or not.

Many detectors use the whole image as an input. These detectors process the input image in all positions, scale etc. Such approaches are computa-

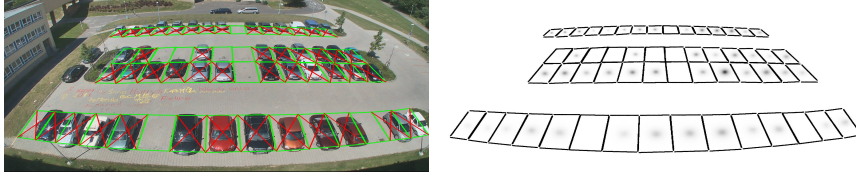


Fig. 3 Accumulated probabilities of car presence over the whole image.

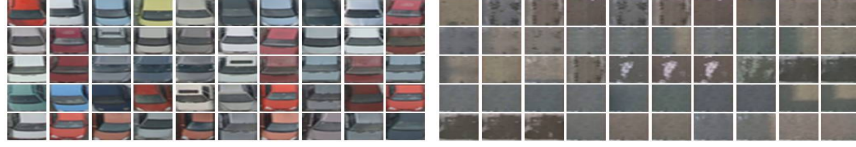


Fig. 4 A sample of positive (left) and negative (right) training samples.

tionally demanding. On the other hand, these approaches allow detection of objects at various locations and sizes. The motivation for the use of our approach with our proposed improvements is to avoid processing the whole image in all different scales and positions in order to accelerate the detection phase of the algorithm while maintaining satisfactory detection results.

5 Experiments

5.1 Training Set

We created a set of training images that contains 4500 positive and 4500 negative samples. The examples of these positive and negative samples are shown in Fig. 4. The positive set consists of cars from the first row (the cars closest to the camera). The cars from the first row are slightly similar to the cars from the back rows (the cars differ mainly in their size between the front and back rows). Furthermore, the samples are not affected by the camera distortion in the first row. The cars in our set were obtained with different climatic and lighting conditions for better detection properties. Moreover, these samples have good image quality and images taken from the first row have the potential to create a data set of a car in the future. Such images could be used in other detection applications. Each positive sample was resized to 20×24 pixels for the training phase.

The negative set of training images was taken from our parking spaces. Each parking space (without cars) was used to the training set. We created this negative set with different climatic and lighting conditions for better detection properties as in obtaining the sample of cars.

Table 1 Detection rates of proposed algorithm and the HOG based detector in noisy images.

Approach	Cars	Det. Cars	Rate	Empty sp.	Det. spaces	Rate	Accuracy
HOG_1	18	17	94.4 %	430	179	41.6 %	43.8 %
Proposed	18	14	77.7 %	430	425	98.8 %	94.2 %

5.2 Real World Tests

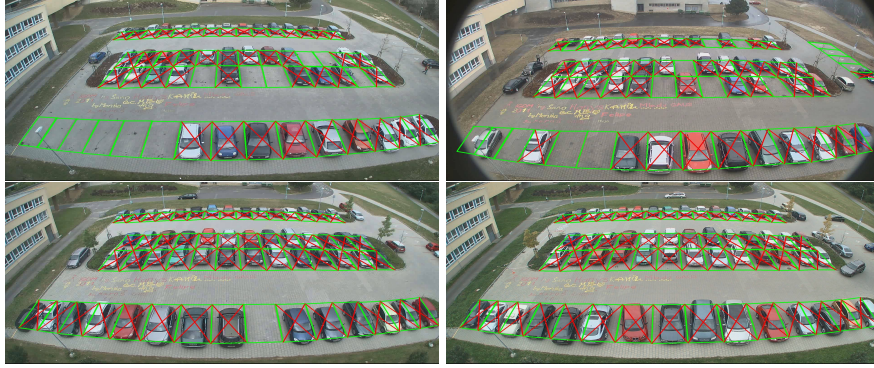
Before continuing to experiments with the proposed detection method, we regard as desirable to describe the selection of this approach instead of detector that is based on the HOG descriptors with SVM. We experimented with the parameters of HOG based detectors and we suggested the optimal configuration (HOG_1). Each positive and negative sample was resized to 96×96 pixels for the training phase of the HOG detector. The configuration of HOG_1 was designed with the size of block = 16×16 , size of cell = 8×8 , horizontal step size = 8, number of bins = 4. This configuration achieved high detection rates in day illumination. Nevertheless, the HOG based detector (with SVM) had the high number of false positive detections in night illumination (in images with noise) than the presented detector. The artifacts that are created by noise have a negative effect to efficient computation of HOG descriptors and the HOG based detector often detects the spaces as occupied although the cars are not in these spaces. The detection rates of detectors (without improvements that we proposed in section 4.1) in noisy images are shown in Table 1.

For the above reasons, we decided to continue to test the presented approach in next experiments only, because we needed a robust detector that is able to detect occupied and vacant spaces in parking lots under day and night illumination with satisfactory results in both illuminations. In order to prove the usefulness of our algorithm in day illumination, we picked up 18 images of park lot in different times of a day. We chose not only images in the cloudy weather, but we also used images in a strong sunlight which causes strong shadows. 9 overcast and 9 sunlight images were chosen, each of them had 56 parking spaces. In total, 1008 parking spaces were examined, 591 were occupied and 417 of them were empty. We show the detection rates with the basic algorithm, algorithm enhanced by moving detection window and shrinking its size in Table 2. This table shows detection results for all test images together (cloudy and sunny weather).

Even the basic algorithm without the enhancements gives nice results with 95.8% correctly detected cars and 90.9% empty spaces. In total, 93.8% of parking spaces were marked correctly. Both numbers can be further improved. If we use the moving detector window, more than 98% of cars were correctly detected, but the detection of empty spaces decreased to less than 89%, even though the accuracy slightly raised. The improvement of car detection rate was achieved by searching for the car multiple times and we detect 14 more

Table 2 Detection rates of algorithm and its enhancements.

Move	Resize	Cars	Det. Cars	Rate	Empty sp.	Det. spaces	Rate	Accuracy
-	-	591	566	95.8 %	417	379	90.9 %	93.8 %
yes	-	591	580	98.1 %	417	370	88.7 %	94.2 %
yes	-2 %	591	578	97.8 %	417	378	90.6 %	94.8 %
yes	-4 %	591	578	97.8 %	417	381	91.4 %	95.1 %
yes	-6 %	591	576	97.6 %	417	387	92.8 %	95.5 %
yes	-8 %	591	563	95.3 %	417	390	93.5 %	94.5 %
yes	-10 %	591	563	95.3 %	417	397	95.2 %	95.2 %
yes	-12 %	591	548	92.7 %	417	401	96.1 %	94.1 %
yes	-14 %	591	550	93.1 %	417	402	96.4 %	94.4 %

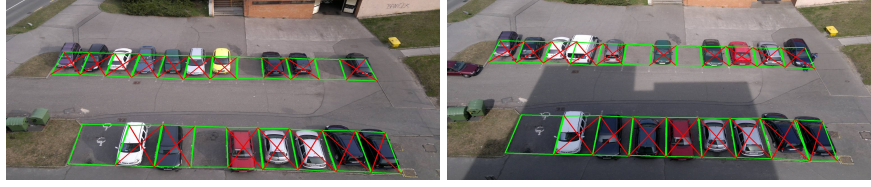
**Fig. 5** Results of detection.

cars from 25 previously undetected. Therefore, the car can be detected multiple times even in situations when it was not detected by the basic algorithm at all. Moreover, we can detect the wrongly parked car more easily. The drawback is that we can also detect the wrongly parked cars in empty spaces by shifting the detector window. In order to improve this issue, we were shrinking the size of detector window. It should still detect the wrongly parked cars in the processed parking space but also should not cover the big part of car from the neighboring parking space. It is obvious that decreasing the size of detector window slowly reduces the car detection rate and slowly increases the empty space detection. Many experiments proved that best results are achieved by shrinking the window by -6 % -8 %. In such cases, the car detection rate is still higher than 97.5 % and empty spaces are correctly detected in slightly more than 95 %.

Table 3 shows the difference between cloudy weather (without shadows) and sunny weather (with problematic shadows). The moving detector window is a significant help in overcast images. The car detection rate rises up to 99 % and we achieved about 96 % to 98 % empty space detection rate with the moving window and the small shrinking factor. The accuracy is approx. 97.5 % to 98 % with the shrinking factor from -2 % to -10 %.

Table 3 Detection rates of algorithm and its enhancements for different conditions.

Move	Resize	Cloudy Weather			Sunny Weather		
		Det. Cars	Det. Spaces	Accuracy	Det. Cars	Det. Spaces	Accuracy
-	-	95.3 %	93.2 %	94.4 %	96.3 %	88.6 %	93.1 %
yes	-	98.0 %	91.7 %	95.4 %	98.3 %	85.8 %	93.1 %
yes	-2 %	99.0 %	96.6 %	98.0 %	96.6 %	84.8 %	91.7 %
yes	-4 %	98.3 %	96.1 %	97.4 %	97.3 %	86.7 %	92.9 %
yes	-6 %	99.0 %	97.6 %	98.4 %	95.9 %	88.1 %	92.7 %
yes	-8 %	97.0 %	98.1 %	97.4 %	93.5 %	89.1 %	91.7 %
yes	-10 %	97.0 %	98.5 %	97.6 %	93.5 %	91.9 %	92.9 %
yes	-12 %	94.6 %	97.6 %	95.8 %	90.8 %	94.8 %	92.5 %
yes	-14 %	95.3 %	98.1 %	96.4 %	90.8 %	94.8 %	92.5 %

**Fig. 6** Results of detection in another parking lot.

On the other hand, the sunny images can be problematic because of strong shadows casted by the neighboring cars to the empty spaces. The car detection rate is still high even in these difficult conditions with 96 % to 98 % using the shrinking factor from -0 % to -6 %. The smaller windows than that quickly reduces the car detection rate. Shadows are a real problem because the empty parking space detection is reduced to 85 % to 88 % with the same parameters and quickly rises up to 95 % with smaller windows. The goal is to choose such shrinking factor that has still the high car detection rate and also has not the very low empty parking space rate. In future work, we are going to try to improve the empty parking space rate in sunny weather by a shadow removal. The result of detections are shown in Figure 5.

We tried to apply our algorithm to the another parking lot with the same classifier. The results are presented in Fig. 6. It is obvious that the trained classifier is sufficient even for the another parking lots. In all three testing images, we achieved 100.0 % detection rate, but we need to say that this parking lot is more simple for the detection algorithm than the former one.

6 Conclusion

We presented the AdaBoost based algorithm for the detection of occupancy of parking spaces provided with the few improvements ensuring the higher robustness of our method. Our test showed that this method provided the

high detection rate of about 98 % in cloudy weather a 93 % in strong sunlight. In future work, we will focus on improvement in sunny images, probably with the shadow removal method.

References

1. Stanley M. Bileschi, Brian Leung, and Ryan M. Rifkin. Component-based car detection. In *in Street Scene Images. S.M. Thesis, EECS, MIT*, 2004.
2. N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, june 2005.
3. Bernd Heisele, Iyaylo Riskov, and Christian Morgenstern. Components for object detection and identification. In *Toward Category-Level Object Recognition*, pages 225–237, 2006.
4. S. Hinz. Detection and counting of cars in aerial images. In *Image Processing, 2003. ICIP 2003. Proceedings. 2003 International Conference on*, volume 3, pages III – 997–1000 vol.2, sept. 2003.
5. S. Hinz, C. Schlosser, and J. Reitberger. Automatic car detection in high resolution urban scenes based on an adaptive 3d-model. In *Remote Sensing and Data Fusion over Urban Areas, 2003. 2nd GRSS/ISPRS Joint Workshop on*, pages 167 – 171, may 2003.
6. Bastian Leibe, Ales Leonardis, and Bernt Schiele. Combined object categorization and segmentation with an implicit shape model. In *In ECCV workshop on statistical learning in computer vision*, pages 17–32, 2004.
7. H. Moon, R. Chellappa, and A. Rosenfeld. Performance analysis of a simple vehicle detection algorithm. *Image and Vision Computing*, 20(1):1 – 13, 2002.
8. Constantine Papageorgiou and Tomaso Poggio. A trainable system for object detection. *Int. J. Comput. Vision*, 38(1):15–33, June 2000.
9. A. N. Rajagopalan, Philippe Burlina, and Rama Chellappa. Higher order statistical learning for vehicle detection in images. In *ICCV*, pages 1204–1209, 1999.
10. Tao Zhao Ram, Tao Zhao, and Ram Nevatia. Car detection in low resolution aerial images. In *Image and Vision Computing*, pages 710–717, 2001.
11. H. Schneiderman and T. Kanade. A statistical method for 3d object detection applied to faces and cars. In *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, volume 1, pages 746 –751 vol.1, 2000.
12. P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, volume 1, pages I–511 – I–518 vol.1, 2001.
13. P. Viola, M.J. Jones, and D. Snow. Detecting pedestrians using patterns of motion and appearance. In *Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on*, pages 734 –741 vol.2, oct. 2003.
14. Tao Zhao and Ram Nevatia. Car detection in low resolution aerial images. *Image and Vision Computing*, 21(8):693 – 703, 2003.