

Scalable real-time parking lot classification: An evaluation of image features and supervised learning algorithms

Marc Tschentscher*, Christian Koch†, Markus König‡, Jan Salmen* and Marc Schlipfing*

*Institute for Neural Computation, Ruhr-Universität Bochum, 44801 Bochum, Germany

Email: marc-philipp.tschentscher@ini.rub.de

†Department of Civil Engineering, University of Nottingham, Nottingham, UK

Email: Christian.Koch@nottingham.ac.uk

‡Chair of Computing in Engineering, Ruhr-Universität Bochum, 44801 Bochum, Germany

Email: koenig@inf.bi.rub.de

Abstract—The time-consuming search for parking lots could be assisted by efficient routing systems. Still, the needed vacancy detection is either very hardware expensive, lacks detail or does not scale well for industrial application. This paper presents a video-based system for cost-effective detection of vacant parking lots, and an extensive evaluation with respect to the system’s transferability to unseen environments. Therefore, different image features and learning algorithms were examined on three independent datasets for an unbiased validation. A feature/classifier combination which solved the given task against the background of a robustly scalable system, which does not require re-training on new parking areas, was found. In addition, the best feature provides high performance on gray value surveillance cameras. The final system reached an accuracy of 92.33 % to 99.96 %, depending on the parking rows’ distance, using *DoG*-features and a support vector machine.

I. INTRODUCTION

Finding a vacant parking spot in urban areas often is a time-consuming and tedious task for drivers [1] and, thus, not satisfying for potential visitors or customers. A system for the detection of vacant parking spaces that routes drivers efficiently to proper lots is desirable. Navigating drivers to lots close to their destination could significantly improve their convenience and leading to a minimized search traffic in urban areas.

Some systems promising to improve the parking lot search and degree of capacity utilization have reached the market or are under research. These systems are described in Sec. II. We review two different types of systems, video-based and other sensor-based systems (like inductive loop detectors, ultra-sound, etc.).

Video-based systems offer a proper alternative to deal with the classification problem and are already established, e.g., for traffic sign detection and classification [2], [3]. Using these systems it is possible to combine low-cost hardware requirements while providing detailed occupancy maps for parking areas. By using video-based systems several challenges occur – especially on outdoor car parks. Different weather and lighting conditions or objects occluding some parking lots might influence the accuracy of the results for the given task.

Section III covers the problems of current systems and describes our main goals: Amongst others, we aim at using gray images only in order to allow using already installed

surveillance cameras. For scalability purposes, it should be possible to easily deploy the system at a new location, i.e., without re-training. As we show in this paper, several image processing and machine learning algorithms, which can be employed to classify vacant and occupied parking lots, already exist. Those algorithms as well as the further system modules are described in Sec. IV.



Fig. 1: View of camera in our experiment

Section V covers the setup and the results of the extensive experiments carried out. We report results on two test datasets including a new parking area (see Fig. 1) which was not utilized for training purposes. Finally, a conclusion from the comparison of image features and machine learning algorithm and from the resulting system’s performance is drawn in Sec. VI.

II. RELATED WORK

A comprehensive overview on the main existing parking space occupancy detection technologies has been previously presented in [4]. Basically, they can be divided into sensor-based (Sec. II-A) and video-based (Sec. II-B) methods.

A. Sensor-based methods

Concerning the sensor installation procedure, the sensor-based systems can be divided into two categories: intrusive and non-intrusive sensor systems. While intrusive sensors are typically installed in the surface, by tunnelling under the surface or anchoring to the surface, leading to invasive installation and maintenance procedures as well as traffic

disruptions, non-intrusive sensors can be easily installed on the ground or the ceiling of a car park. The first kind of systems are also referred to as pavement embedded systems whereas the latter are sometimes called overhead technologies [5].

A common type of pavement embedded sensors are inductive loop detectors (ILDs), which are wire loops either installed at the car park entrance to count entering and leaving vehicles or at each parking space leading to expensive and disruptive maintenance work [6]. Moreover, the loops are subject to wear and tear due to stresses of traffic and temperature, and they are sensitive to water [5]. Another type of embedded systems uses magnetic field sensors, either magnetometers or magnetoresistive sensors (e.g., anisotropic magnetoresistance sensors), that measure changes in the magnetic flux to detect parking vehicles [7]. Although these kind of sensors are insensitive to weather conditions and certain types can be installed on the surface, they need to be employed at each parking lot which can be very costly as each sensing unit is usually attached with a processing unit and a transceiver [8].

Overhead occupancy detection methods are either based on light, sound, or radar sensor systems. Active and passive infrared sensors are usually installed above the vehicles, transmit and receive energy (active) or only receive energy (passive), and recognize changes in the characteristics of the received energy to detect the occupancy status of a parking lot [5]. The drawback of infrared sensors is their sensitivity towards environmental conditions such as heavy rain, dense fog, and blowing snow [4]. On the contrary, sound based sensors, such as acoustic or ultrasonic sensors, which detect sound energy increase and decrease to determine the lot status, are insensitive to humidity. However, large temperature changes and extreme air turbulence negatively affect their performance. Radar sensors, such as microwave radar, perform well in inclement weather conditions, but sometimes need to be equipped with auxiliary sensors to detect stopped vehicles [9]. In general, overhead technologies are difficult to install in large outdoor car parks, which limits their applicability in such environments.

In all cases of sensor based systems the lot occupancy status can be transmitted either over a wired or a wireless network to a remote management center. While long and complicated wiring is becoming obsolete due to extensive installation and maintenance effort, wireless technology is getting the standard implementation [8].

B. Video-based methods

Besides sensor based systems, camera based systems have gathered great attention in recent years [10], since they have the potential to provide a cost effective solution as they allow wide area detection and regular maintenance is possible without disturbing the traffic flow. This substantially reduces life cycle costs and increases detection flexibility [5]. Moreover, they can use existing visual surveillance infrastructure such as security cameras to capture images and videos [11]. Either static images or sequences of images (videos) are fed into computer vision based algorithms that are designed to classify the occupancy of a parking lot as vacant or occupied. Existing vision-based approaches can be roughly classified into three categories: vehicle-driven, space-driven and mixed methods. Table I sums up crucial

properties of each video-based approach.

1) *Vehicle-driven methods*: Under vehicle-driven methods, parking vehicles are the major target and algorithms are developed to detect those. Based on the detection result, vacant parking spaces are determined. A large number of different vehicle detection algorithms have been proposed in recent years.

True [12] has combined interest point detection and color histogram classification using the k-nearest neighbor algorithm and support vector machines to detect vacant parking spaces. The limitations of this work are the relatively low detection accuracy (94%) and the “very slow” processing speed.

Ichihashi et al. [9] have improved the detection performance using fuzzy c-means clustering and hyperparameter tuning by particle swarm optimization. Their system has reached a detection rate of 99.6% for outdoor environments. However, they do not provide detailed information about real-time applicability but mentioning that their approach is “efficient” to join images from multiple cameras.

Tsai et al. [13] have presented a general approach to the detection of vehicles on highways, road intersections, and parking lots under different weather conditions and vehicle orientations. In this approach, a Bayesian classifier based on color features was used to select candidates. Following that, a cascade multichannel classifier based on corner features, edge features, and wavelet coefficients was trained to verify the candidate detection. However, the system is solely based on static images and has reached an average accuracy of merely 94.9%.

Bong et al. [8] have proposed a vehicle detection system that combines image difference, object detection, edge detection, and voting algorithms. Although their approach has been tested under different weather and illumination conditions, it performs moderate in terms of detection accuracy (> 93%).

In previous research, we build a system to classify vacant and occupied parking lots [14]. We run several test on our dataset containing approximately 10,000 samples for training and testing using color information in combination with other image features. We reached an accuracy of 99.8% with real-time processing (> 5 fps) using a support vector machine for the classification process.

2) *Space-driven methods*: For space-driven methods, the characteristics of a vacant parking space are in the major focus such that available vacant parking spaces can be detected directly. For that purpose, common background subtraction algorithms are used that assume statistically stationary background variation. Unfortunately, this assumption might be violated in outdoor scenes, since passing clouds may suddenly change illumination.

Funck et al. [1] have presented a method to handle dynamic variations of an outdoor environment by creating an eigen-space representation that contains a huge set of background models. In this method, the occupancy estimate is determined by the vehicle to car park (empty lot) pixel ratio while compensating perspective distortion and occlusion. However, the high rate of error in the detection 10% can be seen as the major drawback.

Other space-driven methods take advantage of the homogeneous appearance of vacant parking spaces.

For example, Yamada and Mizuno [15] have proposed a

homogeneity measure by calculating the area of fragmental segments. Under their method, a vacant space is assumed to have fewer but larger segments, while a parked car creates the opposite result. Although their system has reached a detection rate of 98.7%, shadows and occlusions caused by adjacent cars are not treated and the processing time is very high (2.7s per image). Similar to [15], Fabin [16] has presented a segment-based homogeneity measure to handle occlusions by training a weighted map to indicate the image regions that may get occupied by neighboring cars. However, results on validation have not been reported.

To reduce perspective effects, Lpez Sastre et al. [17] have suggested to apply a Gabor filter bank on rectified images to derive the homogeneity feature for vacant lot detection. Even though this method has reached an overall classification rate of 99.7%, the experimental setup does not suggest transferability to new areas. Firstly, training and test data were drawn from a common data set of a single parking area. This simplifies the classification task, in particular for the free spots. Secondly, representing only 7.5% of the data, the number of occupied lots is rather small, which puts the reported performance into perspective.

In [18], Wu et al. have presented a multi-class *support vector machine (SVM)* to classify the state of three neighboring spaces as a unit. Markov Random Fields (MRF) have then been used to improve the final classification results by resolving conflicting classifications between two neighboring threespace patches. The performance evaluation reports an error rate of 2.6% with real-time capability.

3) *Mixed and other methods*: Mixed approaches try to combine both vehicle-driven and space-driven methods to improve the detection rate. Huang and Wang [10] have presented a Bayesian detection framework that takes into account both a ground plane model and a vehicle model to represent environmental changes, including perspective distortion, occlusion, shadows and lighting fluctuation. A high accuracy of 99.4% has been reported, but their systems does not reach real-time performance. Lee et al. [19] have suggested a method that combines an entropy-based metric to model the parking space region and a vehicle movement tracker and recorder to detect vehicles. However, their system has been designed for general object identification in an outdoor environment, such as moving pedestrians and cars, but not for vacant space detection in typical parking lots.

Recently, Kabak and Turgut [20] and Seo et al. [21] have presented methods to detect vacant parking spaces using aerial images. Mean-shift image segmentation, followed by a combination of geometrical, optical and statistical feature extraction, and an *SVM*-based classification is used by Kabak and Turgut. Seo et al. have presented a self-supervised learning algorithm that automatically obtains a set of canonical parking spot templates to learn the appearance of a parking lot and estimates the structure of the parking lot from the learned model. For both methods an accuracy of 93.2% and 91.5%, has been reported, respectively. However, real-time applicability has not been in the focus due to the latency in image acquisition.

A different approach to vision-based parking space detection is stereo vision using two cameras. For example, Jung et al. [22] presented a method based on pixel structure classification and feature based stereo matching to extract

3D information in real time. Both the obstacle depth map and the bird’s eye 2D view are used to increase operation speed and robustness. However, this approach requires sufficient overlapping of the two camera views, which prevents it from being practical on large car parks or using already installed surveillance cameras.

III. PROBLEM STATEMENT AND OBJECTIVES

Existing solutions for real-time parking lot classification either rely on embedded and overhead sensors or use camera video streams to determine the occupancy status. Sensor-based systems usually require labor-intensive and time-consuming installation of hardware and wiring for each single parking lot and, in addition, overhead technologies are difficult to install in large outdoor car-parks. These problems prevent the applicability of sensor-based systems in outdoor environments.

Video-based systems, on the other hand, have the potential to provide a cost-effective solution as they support large area observations, do not require tedious sensor installation (except camera sensor) and allow maintenance operation without disturbing the traffic flow. Table I overviews crucial properties of the video-based methods discussed above.

TABLE I: Current state of research. We identified five important properties which are detailed in the text. The cells containing a “?” denote the authors did not report on their system’s processing speed.

	real-time capability (> 5 fps)	high accuracy (> 95%)	source code and dataset publicly available	possible use of existing gray-level surveillance cameras	easy transfer to new outdoor location successfully proved
vehicle-driven methods					
True [12]	-	-	-	-	✓
Tsai et al. [13]	✓	-	-	-	-
Bong et al. [8]	-	-	-	✓	-
Ichihashi et al. [9]	?	✓	(-)	✓	-
Tschentscher et. al. [14]	✓	✓	-	-	-
space-driven methods					
Yamada and Mizuno [15]	-	✓	-	✓	-
Funck et al. [1]	?	-	-	✓	-
Lpez Sastre et al. [17]	?	✓	(-)	✓	-
Wu et al. [18]	✓	✓	-	-	-
Fabin [16]	?	-	-	-	-

Unfortunately, several papers do not report on processing speeds. Nevertheless, solutions that meet the requirement of high detection rates exist, e.g., [9], [14], [17], [18]. Each of those implements individual image features and machine learning algorithms that have been tested on different datasets with various camera perspectives. For this reason, no overall conclusion on the best combination of image features and machine learning algorithms can be drawn.

We argue that datasets and source code should be freely available in order to allow systematic and unbiased comparison of approaches. Only this procedure enables

researchers to reliably reproduce results and compare them to those achieved by their own methods. Therefore, we also consider public availability of these resources as a criterion in Tab. I. Only two studies state the use of free software [9], [17]. Nevertheless, none of the authors published their own resources.

Finally, we want to focus on two additional aspects concerning the practical application in this study to improve our current system [14]:

- It is highly desirable that parking lot classification can make use of available surveillance cameras in order to reduce cost. It should be noted that such cameras typically do not provide color information.
- Applying a system at a new, unseen location should be easy, i.e., not require extensive adjustments like training, parameter tuning etc.

We marked the fulfillment of these two properties in Tab. I for each algorithm. Please note that [9] focused on indoor scenarios while we investigate the more demanding outdoor scenarios in this study.

Therefore, the goal of this paper is to present a video-based outdoor parking lot classification system, which could be easily applied to new parking lots because no re-training is needed and existing surveillance cameras could be used. Moreover, the paper evaluates outdoor experiments comparing several image features and machine learning algorithms supporting highly accurate and real-time capable classification, and finally gives recommendations on their best combination.

IV. METHODOLOGY

In this section the methodology of the classification system is described. Figure 2 shows an overview of the modules that are involved in the classification process for parking lots.

The used methods and algorithms are presented in the following. The system is mainly divided into four parts. The intrinsic calibration (Sec. IV-A) uses a calibration to undistort the distorted image from the wide-angle lens. The preprocessing part (Sec. IV-B) transforms the camera frame into the world coordinates and defines the parking lots. Each parking lot snippet is then feed into the feature extraction part (Sec. IV-C), where different image features are calculated. The classification part (Sec. IV-D) then classifies using the before mentioned features to distinguish between occupied and vacant parking lots and applies a temporal filtering via exponential smoothing to the classification results on single images. At a last step, we implemented a visualization to display the results (Sec. IV-E).

A. Intrinsic calibration

As a first step, we need to find an intrinsic calibration for the camera. Due to the wide-angle lens of the camera we used in our experiments, the images are radial distorted and infeasible for further processing. To get undistorted images, we used the *radial distortion model* as given in [23].

B. Extrinsic calibration and preprocessing

In order to obtain exact information of the observed lots providing a transformation from camera-image to the world-frame is a requirement. Therefore, a direct linear transformation [24] which calculates the required transformation matrix, is employed. Providing at least four world

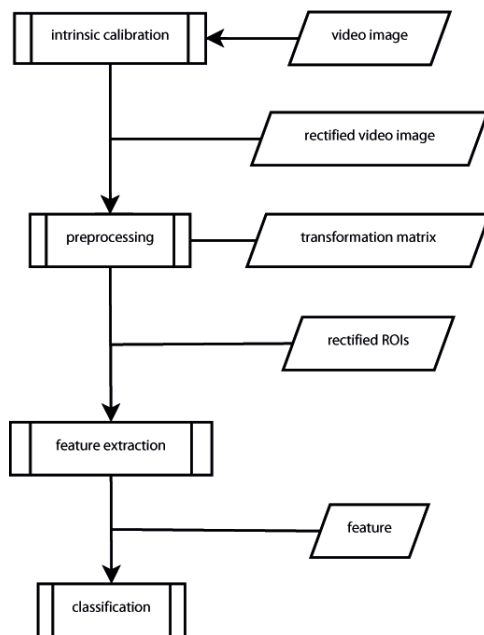


Fig. 2: Overview of the different modules used for classification

coordinates and their corresponding points in the image by hand is a requirement to calculate this transformation.

As a next step, the parking lots that are covered by the camera have to be manually marked in the video image. It is sufficient to mark the four edges of the whole parking row and to provide the number of lots in this area. These labeled *regions of interests* (ROIs) are extracted from the image and scaled to a common size. This is a requirement of the classification. Figure 3 shows examples of such rectified ROIs.



Fig. 3: Rectified ROIs extracted from video image

C. Feature extraction

We implemented several image features to test their ability to describe vacant and occupied parking lots in our previous work [14]. We now focus on color histograms and difference-of-Gaussian (DoG) histograms separately because those features yielded the most promising results in prior tests.

1) *Color histograms*: Color histograms provide detailed information about the distribution of colors in images. In order to receive this information, we developed an extractor that is able to create multi-dimensional histograms with different resolutions per channel. We denote (4, 3, 2) as count of bits for each channel (4 bits for first channel, 3 bits for second channel and 2 bits for third channel).

Figure 4 shows an example for three different color spaces. In this paper, we used RGB, HSV and YUV color spaces because they are widely known and seem promising to us. The RGB color space decodes colors by values of fundamental colors red, green and blue. The HSV and the YUV color spaces have a single lighting channel (V in HSV and Y in YUV). That makes it more usable for different lighting situations as the lighting channel could be ignored on building a classifier. The HSV color space represents colors additionally by their hue (H) and saturation (S) and so forms a cone in which grayish colors shape a closed region. In YUV-color space the two channels U and V encode the color.

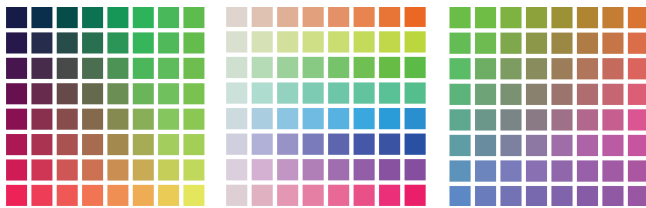


Fig. 4: Example of RGB (left), HSV (center) and YUV (right) color histograms, each showing two components with a resolution of 3 bits.

2) *DoG histograms*: The *Difference-of-Gaussian* method is an image filter detecting edges. Its functionality is similar to a band pass filter. To extract edge information from a camera image, at first, a strongly smoothed image is generated by a *Gauss* filter. This image is then subtracted from the original. The histogram captures the distribution of the absolute values of this difference image. Figure 5 shows examples of *DoG* images.

We particularly investigated gray-value features because we would like to use existing infrastructure, such as surveillance cameras at parking areas, which normally provide monochrome images only.

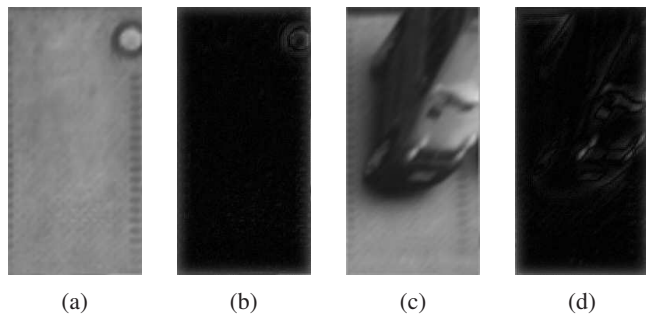


Fig. 5: Source image and output of the *DoG*-feature extractor for vacant ((a),(b)) and occupied parking lot ((c),(d))

D. Classification

In the given scenario, a binary classification of single parking lot samples (occupied/vacant) has to be performed. We applied three supervised learning algorithms, namely *k*-nearest neighbor (*k*-NN) (Sec. IV-D1), linear discriminant analysis (*LDA*) for a linear classifier (Sec. IV-D2) and support vector machine (*SVM*) (Sec. IV-D3). For the implementation of those, the Shark¹ library – a fast, modular, feature-rich open-source C++ machine learning library [25] – was used.

To improve robustness against sporadic misclassification, temporal integration of the frame-by-frame classification results was deployed via *exponential smoothing* (Sec. IV-D4).

1) *K-nearest neighbor*: The *k*-NN classifier is a simple machine learning algorithm. During training, each example's feature vector (containing the selected image feature combination) and its dedicated class label is simply stored. In the classification phase, *k*-NN assigns a test sample to the most frequently occurring class label among its *k* closest training samples in feature space. To prevent overfitting, *k* should be reasonably adjusted. The downside of *k*-NN classification is the complexity, which scales linearly with the number of training examples.

2) *Linear discriminant analysis*: The linear discriminant analysis is a method to derive a linear classification rule [26]. During training, the algorithm finds a hyperplane in feature space in order to separate both classes. The corresponding normal vector can then be used to linearly classify new elements. The procedure, therefore, easily fulfills real-time constraints. The used model assumes the data of both classes to be normally distributed around the class centers with a common covariance. Despite this simplicity, *LDA* gives surprisingly good results in practice.

3) *Support vector machine*: Amongst others, *SVMs* [27] are used to classify non-linearly separable data sets. The input feature vectors are transformed into a higher dimensional space during training. The algorithm optimizes a hyperplane, trying to separate the labeled training elements of the two classes from each other with a possibly large safety margin. This hyperplane is represented by those examples lying on or beyond the margin, the so-called support vectors. Non-linearity is introduced through a kernel expansion, where the dot products of a sample with each support vector is replaced by a kernel function. Therefore, an *SVM* is able to build more flexible decision boundaries in input space while preserving good generalization properties.

4) *Temporal integration*: For temporal integration exponential smoothing was applied to the classification results. Equation 1 shows the filters equation where α represents the learning rate and Y represents the particular class label (vacant/occupied) for each parking lot. Setting $\alpha = 0$ no temporal integration will be performed meaning that old classification results do not affect the current result. By adjusting $\alpha > 0$ the classification results gets robust but the detection for a change of the occupancy of a parking lot will be time delayed. If $\alpha = 1$, no new classification

¹<http://image.diku.dk/shark>

will affect the older for a specific parking space. The term $f(t)$ is calculated for every new classification of a parking space and, leads to a temporal integrated probability of a vacant space.

$$f(t) = \alpha \cdot Y_{t-1} + (1 - \alpha) \cdot Y_t \quad (1)$$

Though using this filter may lead to a time-shifted (i.e. slightly delayed) detection, this circumstance is insignificant for implementing it for car park routing systems.

E. Visualization

The visualization of the current occupancy of the parking area is drawn directly in the transformed image. There is one image per parking row where unoccupied lots are marked with a green border and occupied with a red border (Fig. 6). Please note that the car parking leftmost in the bottom row occupies two lots.

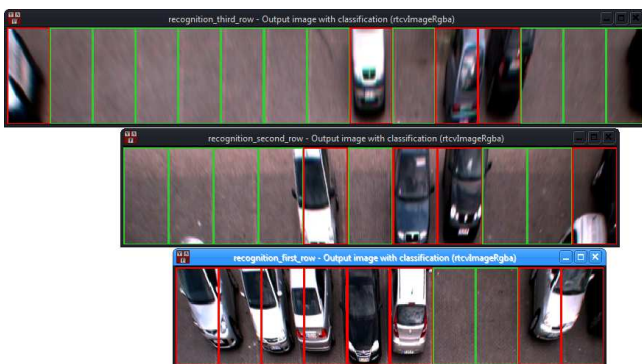


Fig. 6: Visualization of the classification occupied (red) and vacant (green) parking lots of the proposed system

V. EXPERIMENTS

This section shows the description of our experimental setup (cf. Sec. V-A) and the results of our test scenarios, including a parking area that was not investigated during training (cf. Sec. V-B).

A. Setup

We provide information about the hardware and environmental setup in Sec. V-A1. The datasets that were used for training and testing different classification methods is described in Sec. V-A2. Section V-A3 explains the parameters that we used for both, the feature extractors and machine learning algorithms.

1) *Hardware and Environmental Setup:* The software runs on a standard desktop PC. We used a laptop equipped with *Intel Core i5-3210M* Processor and 6 GB working memory running Microsoft Windows 7.

In order to cover a possibly large amount of parking spaces, an off-the-shelf camera with a wide-angle lens is used for recording. Similar to a surveillance camera, it is positioned in the back of the parking lots with a slight top view of 7.5 m (see Fig. 1).

In our configuration, we are able to observe the first three parking rows with a high reliability. The fourth row could not be observed because cars parking right in front can occlude parking lots completely. This leads to a monitored area of 36 parking lots per camera. This fact

keeps the hardware and installation costs rather low, given that otherwise a single sensor has to be applied to each lot.

2) *Datasets:* We recorded video sequences at two different parking areas on three different days:

- 1) Parking area “ID”, sunny
- 2) Parking area “ID”, different lighting conditions, foggy
- 3) Parking area “FH”, typical autumn day

The status of all relevant parking lots was labeled manually throughout all sequences. Then, we built three independent datasets based on the scenarios specified above:

- A. Training: 5,000 snippets from scenario 1
- B. Test: 5,000 snippets from scenario 2
- C. Validation: Full sequence (2.5 min) from scenario 3

For the first two datasets, we ensured that the numbers of snippets for both classes (occupied vs. vacant parking lots) are fairly balanced in order to avoid biased results (cf. Sec. II). Figure 3 shows examples from both, training and test dataset. For training and testing, only those parking lots that could be clearly labeled were considered, that is, we discarded partly covered lots for instance. The validation sequence C comprises 10,090 snippets for the first parking row, 11,110 snippets for the second, and 15,150 snippets for the third row. An example of a video image can be found in fig. 1.

The first dataset was used for training different combinations of feature extractors and classifiers (see below). Then the test dataset was used for selecting those approaches showing the best performance on unseen data, i.e., generalize well. The validation dataset was finally applied to evaluate the resulting system as a whole (including temporal integration, cf. Sec. IV-D4). These results allow for reliable implications on the practical applicability of the system as it is tested on an unseen location.

All datasets used in this study will be made publicly available for free use on request in order to test own algorithms and compare to given results.

3) *Parametrization:* We used three different color spaces and varying channel resolutions for the color histograms. The *DoG*-Feature extractor was tested using different filter sizes (3×3 , 5×5 , 7×7 , 9×9 , 11×11 , 13×13 , 15×15 and 17×17) for the *Gauss* filter. The machine learning algorithms were parametrized as follows: We set $k = 5$ and used 50 prototypes per class using the k -NN classifier because we experienced that this is a good compromise between robustness and processing speed to get reliable classification results. The *LDA* was regularized with $\sigma^2 \in [10^{-6}, 1]$. The *SVM* parameters during training were selected from $\gamma \in [0.1, 100]$ and $C \in [0.01, 1000]$, using a radial basis function kernel.

We used 3-fold cross-validation to select a model for *SVM* and *LDA* classifier. The training data is splitted randomly in three disjoint parts, two parts for training and the other for validating the classifier. This process is repeated two times, using all possible combinations for training and validation, respectively. The average of all three evaluations is used for estimating the quality of the classifier. Cross-validation allows finding a classifier that generalizes well.

Concerning the temporal filtering we chose $\alpha \in [0.7, 0.8]$ and $f(t) > 0.8$ as a threshold.

B. Results

In this section we present the results for the test dataset (Sec. V-B1) and the validation dataset (Sec. V-B2) for evaluating the portability of each classifier-feature combination.

1) *Test results:* We considered variations of different features (Sec. IV-C) in combination with three machine learning algorithms (Sec. IV-D). Each combination was trained on dataset A and then tested on dataset B (cf. Sec. V-A2). Tables II and III show the classification results on the test dataset for color histogram features and *DoG* histogram features respectively.

TABLE II: Classification results for single images on test dataset using color histogram features. The best performing features are highlighted.

feature	k-NN	LDA	SVM
RGB (2,2,2)	96.35	94.59	97.12
RGB (3,3,3)	92.72	92.69	93.20
HSV (0,3,3)	94.68	94.39	95.69
HSV (3,3,0)	95.46	94.09	96.08
HSV (3,3,2)	95.25	94.15	95.90
HSV (4,0,0)	96.55	94.24	96.35
HSV (4,4,0)	95.72	95.52	97.09
YUV (0,4,4)	94.87	88.06	93.44
YUV (2,2,2)	96.55	94.36	95.19
YUV (3,1,1)	95.37	94.03	96.79

TABLE III: Classification results for single images on test dataset using *DoG*-features. The best performing features are highlighted.

feature	k-NN	LDA	SVM
3×3	87.82	83.22	91.98
5×5	87.91	82.57	92.35
7×7	87.26	83.31	92.12
9×9	86.93	80.04	94.13
11×11	91.83	80.10	94.02
13×13	93.11	81.50	93.91
15×15	93.26	82.66	93.08
17×17	93.58	82.03	93.61

Based on the test results, we chose the best performing feature extractor parameters for each machine learning algorithm. As we chose the feature/classifier combinations based on the independent test set, we can assume that these solutions generalize well and, therefore, allow for good results also on other unseen data.

2) *Validation results:* Finally, we tested the performance of the selected methods on dataset C which originates from another parking area to evaluate the portability of each classifier-feature combination. For this experiment, we evaluated results on a full sequence and, therefore, incorporated temporal integration (see Sec. IV-D4). The initialization of the temporal filtering needs several frames before robust classification results can be derived. Thus, the first 3 seconds are discarded for the evaluation.

Tables IV and V show the results obtained with color features and *DoG* features respectively. We distinguish between the three parking rows (1st row nearest, 3rd row furthest).

TABLE IV: Classification results for a sequence of the unseen parking area using color histogram features

class.	feature	1st row	2nd row	3rd row
k-NN	HSV (4,4,0)	96.68	98.95	91.83
	YUV (2,2,2)	79.55	86.59	87.13
LDA	HSV (4,4,0)	61.40	74.97	72.26
SVM	RGB (2,2,2)	72.67	73.66	69.23

TABLE V: Classification results for a sequence of the unseen parking area using *DoG*-features

class.	feature	1st row	2nd row	3rd row
k-NN	17×17	89.96	90.32	90.48
LDA	7×7	81.26	96.32	37.78
SVM	9×9	99.96	96.99	92.33

C. Discussion

It is noteworthy that – depending on the learning method considered – very different features led to the best performance on the test dataset, i.e., different color spaces and different filter sizes. Nevertheless, color-based features always outperformed the *DoG* features based on gray images.

On the validation sequence (dataset C), the performance of color-based *LDA* and *SVM* dropped significantly. This does not hold for the *DoG* features which, therefore, seem to support good generalization.

Combinations where the distance of the parking cars to the camera does not affect classification results exist (17×17 *DoG* features with *k-NN*). In contrast, for some methods the distance has strong impact (7×7 *DoG* features with *LDA*).

In summary, we found a very robust solution that allows for high performance, namely 9×9 *DoG* features in combination with an *SVM* classifier. Interestingly enough, this approach can be used with existing surveillance cameras as no color information is needed.

VI. CONCLUSION

In this paper, we extended our video-based system [14] for vacant parking space classification using image features and machine learning algorithms. The main difference to [14] is the use of single features in contrast to feature combinations in order to reduce computing time and investigate the use of already installed surveillance cameras. We analyzed several different features with various color histograms or *DoG* histograms and three supervised learning algorithms (*k-NN*, *LDA*, *SVM*) and evaluated their performance on a recorded dataset that will be made freely available. The system was validated on a new parking area and reached very high accuracy without any re-training. This verifies the system to be scalable and robustly transferable to new car-park scenarios without the need for cumbersome data collection, annotation, and training.

The final system relies on *DoG* features with filter size of 9×9 , an *SVM* classifier, and *exponential smoothing* for temporal filtering. Installed at the unseen parking area, it reached an accuracy of 99.96% at the first parking row and achieved real-time speed (5 fps) for the classification of 36 parking lots in a single video image into *vacant* or *occupied*. This leads to a classification speed of 179.86 lots per second since the processing speed of calculating *DoG* feature and classifying via *SVM* for each parking lot image is the same. The third parking row is still classified with an accuracy of 92.33%. As the *DoG* features do not require color information, they support the use of monochrome images from existing surveillance cameras.

We plan to use this system in parking garages even though it was only tested on outside parking lots. Due to the lack of space, more cameras would be needed to cover a large range of lots. Furthermore, the system's parameters have to be adopted to changed lighting conditions to gain a sufficient performance.

Further improvements can be achieved by minimizing the influence of adjacent cars parking left and right of the regarded parking lot. This already poses a problem on the current setup where the camera distance is still very large. Therefore, we aim at a 3D-estimation of parking cars to obtain the occluded space of nearby parking lots.

Additionally we currently work on an embedded system. It will make it possible to create *smart cameras* which are already equipped with a small processing unit to calculate the occupancy map directly. This will enhance the portability of our system because this approach only needs a *WiFi* connection to provide the occupancy map to a server. A smart-phone application will then visualize the current car park situation.

REFERENCES

- [1] S. Funck, N. Mhler, and W. Oertel, "Determining car-park occupancy from single images," in *Proceedings of the IEEE Intelligent Vehicles Symposium*, 2004, pp. 325–328.
- [2] Y. Wuept, Y. Liu, J. Li, H. Liu, and X. Hu, "Traffic sign detection based on convolutional neural networks," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2013, pp. 1–7.
- [3] D. Ciresan, U. Meier, J. Masci, and J. Schmidhuber, "A committee of neural networks for traffic sign classification," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, 2011, pp. 1918–1921.
- [4] M. Y. I. Idris, Y. Y. Leng, E. M. Tamil, N. M. Noor, and Z. Razak, "Car park system: A review of smart parking system and its technology," *Information Technology Journal*, vol. 8, no. 2, pp. 101–113, 2009.
- [5] K. C. Mouskos, M. Boile, and N. Parker, "Technical solutions to overcrowded park and ride facilities, university transportation research center region 2," City College of New York, Final Report FHWA-NJ-2007-01, 2007.
- [6] T. Ristola, "Parking guidance system in tapiola," in *Proceedings of the IEEE Conference on Road Traffic Monitoring*, 1992, pp. 195–198.
- [7] J. Wolff, T. Heuer, H. Gao, M. Weinmann, S. Voit, and U. Hartmann, "Parking monitor system based on magnetic field sensors," in *Proceedings of the IEEE Conference on Intelligent Transportation Systems*, 2006, pp. 1275–1279.
- [8] D. B. L. Bong, K. C. Ting, and K. C. Lai, "Integrated approach in the design of a car park occupancy information system (COINS)," *IAENG International Journal of Computer Science*, vol. 35, no. 1, pp. 7–14, 2008.
- [9] H. Ichihashi, T. Katada, M. Fujiyoshi, A. Notsu, and K. Honda, "Improvement in the performance of camera based vehicle detector for parking lot," in *Proceedings of the IEEE International Conference on Fuzzy Systems*, 2010, pp. 1950–1956.
- [10] C.-C. Huang and S.-J. Wang, "A hierarchical bayesian generation framework for vacant parking space detection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1770–1785, 2010.
- [11] A. Nallamuthu and S. Lokala, "Vision based parking space classification," Clemson University, Department of ECE, Report, 2008.
- [12] N. True, "Vacant parking space detection in static images," University of California San Diego, Report, 2007.
- [13] L.-W. Tsai, J.-W. Hsieh, and K.-C. Fan, "Vehicle detection using normalized color and edge map," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 850–864, 2007.
- [14] M. Tschentscher, M. Neuhausen, C. Koch, M. Knig, J. Salmen, and M. Schlopsing, "Comparing image features and machine learning algorithms for real-time parking-space classification," in *Proceedings of the ASCE International Workshop on Computing in Civil Engineering*, 2013, pp. 363–370.
- [15] K. Yamada and M. Mizuno, "A vehicle parking detection method using image segmentation," *Electronics and Communication*, vol. 84, no. 10, pp. 25–34, 2001.
- [16] T. Fabin, "An algorithm for parking lot occupation detection," in *Proceedings of the IEEE Conference on Computer Information Systems and Industrial Management Applications*, 2008, pp. 165–170.
- [17] R. J. Lopez-Sastre, P. G. Jimenez, F. J. Acevedo, and S. M. Bascon, "Computer algebra algorithms applied to computer vision in a parking management system," in *Proceedings of the IEEE International Symposium on Industrial Electronics*, 2007, pp. 1675–1680.
- [18] Q. Wu, C. Huang, S.-Y. Wang, W.-C. Chiu, and T. Chen, "Robust parking space detection considering inter-space correlation," in *Proceedings of the IEEE International Conference on Multimedia and Expo*, July 2007, pp. 659–662.
- [19] C. H. Lee, M. G. Wen, C. C. Han, and D. C. Kuo, "An automatic monitoring approach for unsupervised parking lots in outdoor," in *Proceedings of the IEEE International Conference on Security Technology*, 2005, pp. 271–274.
- [20] M. O. Kabak and O. Turgut, "Parking spot detection from aerial images," Stanford University, Final Project Autumn 2010, Machine Learning class, 2010.
- [21] Y.-W. Seo and C. Urmson, "Utilizing prior information to enhance self-supervised aerial image analysis for extracting parking lot structures," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, October 2009, pp. 339–344.
- [22] H. G. Jung, D. S. Kim, P. J. Yoon, and J. H. Kim, "Stereo vision based localization of free parking site," in *Computer Analysis of Images and Patterns*, ser. Lecture Notes in Computer Science, A. Galalowicz and W. Philips, Eds. Berlin / Heidelberg: Springer, 2005, vol. 3691, pp. 231–239.
- [23] J. Weng, P. Cohen, and M. Herniou, "Camera calibration with distortion models and accuracy evaluation," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 14, no. 10, pp. 965–980, 1992.
- [24] Y. Abdel-Aziz and H. Karara, "Direct linear transformation from comparator coordinates into object-space coordinates in close-range photogrammetry," in *Proceedings of the ASP/UI Symposium on Close-Range Photogrammetry*, 1971, pp. 1–18.
- [25] C. Igel, V. Heidrich-Meisner, and T. Glasmachers, "Shark," *Journal of Machine Learning Research*, vol. 9, pp. 993–996, 2008.
- [26] T. Hastie, J. Friedman, and R. Tibshirani, *The elements of statistical learning*. Springer Series in Statistics, 2001.
- [27] V. N. Vapnik, *Statistical Learning Theory*. Wiley-Interscience, 2008.