

Texture and Color Descriptor Features-based Vacant Parking Space Detection using K-Nearest Neighbors

A F M Saifuddin Saif¹, Zainal Rasyid Mahayuddin²

Department of Computer Science and Information Systems,
West Virginia University Institute of Technology, West Virginia, USA¹
Faculty of Information Science and Technology,
Universiti Kebangsaan Malaysia, 43600 UKM, Bangi, Selangor, Malaysia²

Abstract—The importance of the detection of vacant parking spaces is increasing gradually. A system capable of detecting vacant parking spaces in real-time can play an important role in saving valuable time for motorists, decreasing traffic jams, and reducing air pollution. Vision-based parking space detection methods are advantageous in terms of installation and maintenance as existing security cameras in a parking area can be used without the requirement of additional hardware and the detection program can be run on a local or a remote server. One major problem of the vision-based detection method in this context is making the model generalized for detection in various weather conditions. This research proposes a hybrid method to detect vacant parking spaces that use texture and color descriptors. A weighted KNN is used for the classification of parking spaces. The proposed method experimented on PKLot, a large benchmarking dataset that contains images of three parking areas in three weather conditions. The proposed model achieves an accuracy of 99.47% on average while training with 10-fold cross-validation and achieves an accuracy of 99.41% accuracy on average while testing with unseen data. The model shows robustness and better performance in terms of accuracy and processing speed. Several comparisons are also done to show how well it performs with methods found in previous research.

Keywords—Texture; color descriptor; k-nearest neighbors; computer vision; image processing

I. INTRODUCTION

The requirement for vehicle parking space is increasing due to the increasing number of people owning vehicles causing unbalance in demand and supply of parking spaces. In addition, another underlying reason is the number of parking spaces in a certain location is fixed but the demand for parking spaces varies at different times of the day. People may not be able to locate parking spaces even after spending significant time looking for one during rush hours. Utilization of the available parking space efficiently can be a key solution in this regard where motorists will need real-time information of the available parking spaces of the nearby parking areas. In this context, the necessity of automatic vacant parking space detection is a demanding research topic in the field of computer vision domain. Vision-based automatic vacant parking space detection can help to find available parking spaces at certain locations using images from preinstalled cameras in a parking area.

Previous researchers proposed various approaches to solve vacant parking space detection, i.e., various sensors-based

detection [1, 2], counter-based detection [3, 4], and vision-based detection [5, 6, 7]. Wireless sensors were previously used to develop parking monitoring systems but the problem with the sensor-based method is that deployment and maintenance of sensors are required also the sensors add additional costs [2]. Counter-based methods keep track of the number of vehicles entering and exiting the parking area, however, counter-based methods cannot provide information about parking space location which cannot be used for a wide range of applications [4]. Although previously some research criticized the vision-based approaches technique because of expensive camera equipment and the need to deal with a large amount of data [6], a vision-based vacant parking space detection system can easily be deployed using existing security cameras of the parking space without additional hardware requirements in parking area [7]. This research focused on detecting vacant parking spaces using texture analysis based on Gray Level Co-Occurrence Matrix (GLCM) also known as Grey Tone Spatial Dependency Matrix and RGB color descriptors. The proposed method was validated using the PKLot dataset [7] which is a large dataset of three parking areas with different weather conditions and different times of the day. PKLot dataset was previously used for validating various previous methods, i.e. CarNet [8], mAlexNet [9], and Deep CNN [10] which are compared with the proposed method based on various performance metrics.

The rest of the research paper is structured as follows: Critical previous research is illustrated in the background study, mentioned in Section II, comprehensive details of the proposed methodology are elaborated in Section III, details of experimental results with analysis for experimental validation are demonstrated in the experimental results and analysis in Section IV and finally, the conclusion in Section V presents concluding remarks.

II. BACKGROUND STUDY

Several research has been done to solve the parking space detection problem while keeping three factors in consideration, i.e., mind robustness, deployment effort, and maintenance cost. Previous research has been conducted resulting in different methods i.e., multicamera vehicle detection [11], drone-based and aerial image analysis [12, 13], image descriptor-based [14], geometric features-based [15], edge-based [16], plane-based [17], convolutional neural network [18, 19], sensor network [2] [20]. Some of the previous research was done

based on wireless sensor networks. Sensors are being placed in parking spaces to transmit data to a server to detect if space or unoccupied is occupied [2] [20]. However, existing related methods related to sensors, cost, and maintenance were not feasible in large parking areas. Currently, researchers are more focused on vision-based parking space detection where images of the parking area are taken using the camera. Research in [21] used two types of approaches for image-based systems, one of them was car-driven and another one was space-driven. In the car-driven approach, methods were developed to detect cars as objects as the main objective. However, the problem with this approach was that when a camera was placed at an angle, images of vehicles near the camera and far from the camera have a significant difference in quality as the number of pixels will be less for far images. So, the detection of objects or vehicles became complex. In the space-driven approach, the detection of space was the main objective. The space-driven approach was less complex than the car-driven approach because space has more similarity than vehicles at a particular place. However, the challenge of the space-driven approach was to create robust methods for different parking areas under different scenarios.

Research in [15] proposed a geometric feature-based method to detect vacant parking space where a line segment was used to detect an algorithm for creating a line-clustering method consisting of several parallels for separating lines with a fixed distance and one guideline. The false line was removed and then the guideline was detected using a learning-based method. This method performed better than the single bird's eye view method proposed by research in [15]. However, this method was unable to meet real-time processing. Research in [13] used aerial images and line detection and combined selective filtering calculated using the prevalence of line length and angle. Their algorithm aimed to do automated detection of parking space regions in parking lot images for collecting parking occupancy information. The advantages of this method have four aspects, i.e., provided well enough results for automatic region extraction, fast, covered a large area, and can be used for automatic segmentation. However, some lines were not detected by that method due to light and shadow variance. Research in [22] used a deep convolutional network by introducing an architecture called "Siamese architecture" for learning robust features of the parking spaces for eliminating inter-object occultation and increased performance in various illumination conditions. They also used three space input patches of a single parking space and trained the network for the classification of parking occupancy. The method proposed by research in [22] was better than the method proposed by research in [13] due to the usage of a drone coupled with a line detection algorithm to detect parking space which was not robust, and the deployment and maintenance would be more difficult than just using single or multiple fixed place cameras. In addition, research in [22] performed better in different illumination conditions wherein research in [13] method of light and shadow variance caused problems in line detection.

Research in [11] proposed a method based on dilated convolution neural networks. They claimed their model to be more robust than other methods proposed by research in [14]

and [16]. Research in [8] used the dataset that was created by research in [2] named PKLot and compared the results with other research methods. There was a significant difference when the dataset was trained in one parking lot and tested in other parking lots. Results by research in [5] showed that their method was more robust as their accuracy did not fall like previous research while training and testing in different datasets. In this context, research in [7] used two texture-based features, i.e., local binary patterns and quantization of the local phase. Support Vector Machine (SVM) was trained in their research to detect vacant parking spaces and received an accuracy of 99% when the model was trained and tested on the same dataset. However, for other datasets, they achieved the best accuracy of 89% using textural features. The method proposed by research in [5] performed better than research in [7] in terms of robustness. Experimental results by research in [8] showed that their proposed method accuracy did not fall like research in [7] while training and testing in different datasets.

Existing research methods need to be robust for parking space detection in terms of various illumination and environmental conditions like sunny, cloudy, and rainy. In addition, these methods need to be easily deployable, and maintenance should be cost-effective. The method proposed by research in [15] was able to provide real-time parking space detection. However, that method cannot be used for a large number of parking areas because outdoor parking areas contain a large number of parking spaces. Research in [13] used aerial images for the detection but using a drone is not cost-effective and not suitable for daily use as the drone has a very limited battery life and high maintenance cost. Research in [22] used fixed-place cameras to detect vacant parking spaces, they used three space input patches of a single parking space detection using CNN and were able to achieve good accuracy. However, research in [7] used only a single patch and texture-based features for the detection which claimed to be performed better compared with research in [22]. Research in [7] and research in [18] lack in terms of robustness because despite providing good accuracy on a single parking area, their accuracy decreased when multiple parking areas were used in the scenario. This research proposes an efficient method for parking space detection using texture and color-based features to make the validation robust. In addition, the proposed method is easily deployable and has low maintenance costs as it can be deployed using the camera that already exists in parking areas.

III. PROPOSED METHODOLOGY

The proposed method aims to detect vacant parking spaces in different weather and illumination condition of the day shown in Fig. 1. The proposed method mainly focuses on the extraction of features that reflect the difference between unoccupied and occupied parking spaces. The overall methodology consists of four steps, i.e., acquisition of images for processing, segmentation of parking spaces using a fixed mask and preprocessing segmented images, color descriptors-based feature and texture-based feature extraction, and detection of parking spaces using supervised machine learning algorithm k-nearest neighbors. The proposed method by this research is depicted in Fig. 1.

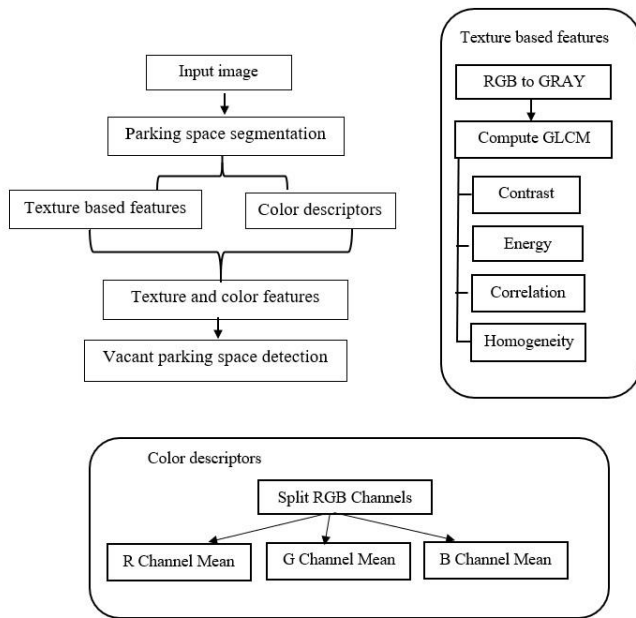


Fig. 1. Proposed method.

A. Input Image

This research used the PKLot dataset of three different parking areas. According to research in [7], a single camera covering the whole parking area was enough for the detection of vacant parking spaces. The dataset was annotated with information of the locations of parking spaces and occupancy status. In terms of real-life use, camera calibration was followed as research in [4] [23]. PKLot Dataset consists of three subsets for three parking spaces, i.e., PUCPR, UFPR04, UFPR05, and images of three different weather conditions sunny, cloudy, and rainy for each parking area. Sample images of the PKLot dataset are mentioned in Table I.

B. Parking Space Segmentation

The parking area may consist of many parking spaces. For this reason, automatic segmentation of parking space will produce extra computational overhead and causes the process not suitable for real-time use. For faster segmentation, a fixed mask once for all spaces was created manually as research in [7] and [24] did in their research which leads them to achieve very fast segmentation of parking spaces. The fixed mask uses the coordinates of the parking spaces. After placing the camera, coordinates of the parking spaces were collected to crop out the patches of the parking spaces from images. The terminology is that for a fixed camera, the parking spaces are static, only a vehicle will move into a parking space or will move out of one. Two copies of a segmented parking space were created for two types of feature extraction, i.e., color feature-based and texture feature-based extraction. For color-based feature extraction in preprocessing part, a segmented RGB parking space is split into three channels R, G, and B as the intensity of the pixels also varies on different channels. The R, G, and B channels of occupied parking space have more variations than the unoccupied parking space, also there is a significant difference between occupied and unoccupied R, G, and B channels, and can be used as features to distinguish

between occupied and unoccupied parking spaces. After color-based features, texture-based features were extracted for the highly distanced parking spaces from the camera which are smaller than less far parking spaces from the camera in addition, highly distanced parking make the structural or geometrical features extraction more complex and not effective but texture-based features can be used to find the similarities and dissimilarities between occupied and unoccupied parking spaces in that scenario. For texture-based feature extraction, images were converted RGB to grayscale followed by median filtering. Details of features extraction regarding color features extractions and texture features extractions are explained comprehensively in the next section.

TABLE I. SAMPLE IMAGES IN DATASETS

	PUCPR	UFPR04	UFPR05
CLOUD Y			
RAINY			
SUNNY			

C. Color Descriptors and Texture-based Features

This research used texture features-based information of the parking space which were collected from the Gray Level Co-Occurrence Matrix and combining them with color-based feature helped to achieve more accuracy and even if the parking space was near or far from the camera, they had similar values and leading to more robust methodology comparing with the previous research. In this section, color descriptors-based features and texture based-features were extracted. For color descriptors-based features, there is a significant difference in occupied and unoccupied RGB channels. Using Eq. (1), the mean value of each channel is calculated to estimate significance difference in occupied and unoccupied RGB channels.

$$\bar{k} = \frac{1}{z} \sum_{j=1}^z k_j \quad (1)$$

where, \bar{k} is the mean of a channel and z is the number of pixels and k_j is the value of the intensity of a pixel. For texture-based features, Gray Level Co-Occurrence Matrix (GLCM) was used to derive some statistical values of the images. The GLCM is constructed from gray images. GLCM

has rows and columns that are equal to the number of tones or gray levels in a grayscale image. In addition, GLCM calculates how often a gray tone or intensity occurs with the adjacent pixels from the input image.

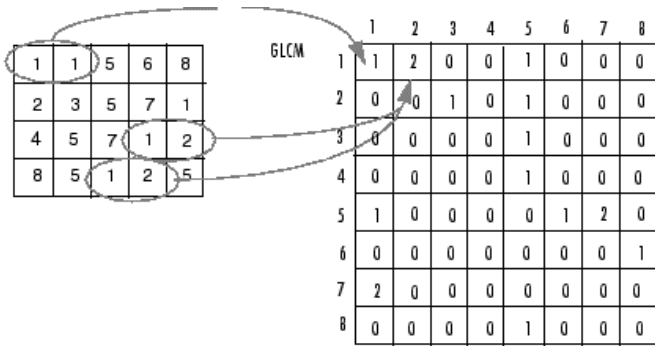


Fig. 2. GLCM construction example for 8-tone grayscale [25].

Fig. 2 shows the calculation of GLCM of 8 tone grayscale images, but in this research, GLCM was calculated for 256-tone grayscale images. GLCM can extract certain texture properties from the spatial distribution of the gray image. In the proposed method by this research, four statistical texture features were computed from the GLCM matrix, i.e., Contrast, Correlation, Energy, and Homogeneity. Various texture features were extracted based on Eq. (2) to Eq. (5) [26].

$$C = \sum_{m,n=0}^{L-1} Q_{m,n}(m-n)^2 \quad (2)$$

$$D = \sum_{m,n=0}^{L-1} Q_{m,n} \frac{(m-\mu)(n-\mu)}{\mu^2} \quad (3)$$

$$E = \sum_{m,n=0}^{L-1} (Q_{m,n})^2 \quad (4)$$

$$H = \sum_{m,n=0}^{L-1} \frac{Q_{m,n}}{1+(m-n)^2} \quad (5)$$

where, L = number of intensity levels, Q_{mn} = element at (m,n), μ = mean of GLCM, σ^2 = intensity variance. Thus, the proposed method dealt with multiple features which were extracted and used for the detection of vacant parking spaces by manipulating the correlation among these multiple features.

D. Vacant Parking Space Detection

The proposed method extracted features represented as numeric values collected from many images from subsets of the PKLot dataset with three different weather conditions. Several classifiers, i.e., logistic regression, Support vector machine, k-nearest neighbor, weighted k-nearest neighbor, and linear discriminant were tested to check how these classifiers perform on the features extracted by this research. Among all the classifiers, the weighted k-nearest neighbors algorithm worked well for the classification of parking space. The parameters used for the model are shown in Table II.

Due to better performance and low complexity, this research used a weighted K-nearest neighbor as the classifier. As all the extracted features are numerical using KNN which

makes training and testing the model less complex. The weighted KNN model was trained and tested with the features that were extracted using the proposed methodology. As the supervised machine learning algorithm needs labeled data, vacant information was collected from the PKLot Dataset [7]. Classified parking spaces were marked as green for unoccupied ones and red for occupied ones shown in Fig. 3 as a result of the proposed method.

TABLE II. PARAMETERS USED FOR WEIGHTED KNN

Parameters	Value	Formula
Distance Metric	Euclidean	$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$
Distance Weight	Squaredinverse	$w = 1/d^2$
Break ties	Smallest	-
Number of Nearest Neighbors	10	-
Nearest Neighbor Search Method	kdTree	-

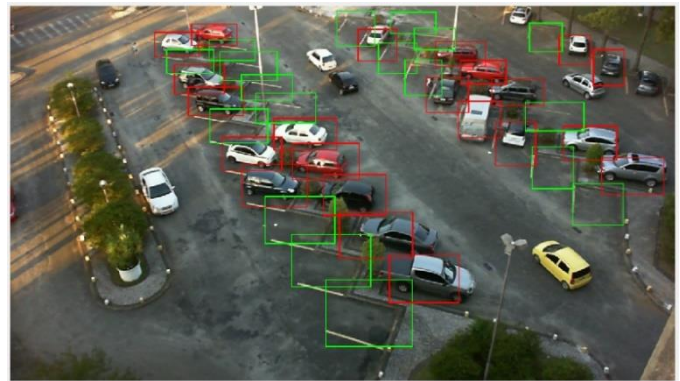


Fig. 3. Classified parking spaces UFPR05-Sunny.

IV. EXPERIMENTAL RESULTS AND DISCUSSION

This research was validated using several phases during experimentation shown in Fig. 4.

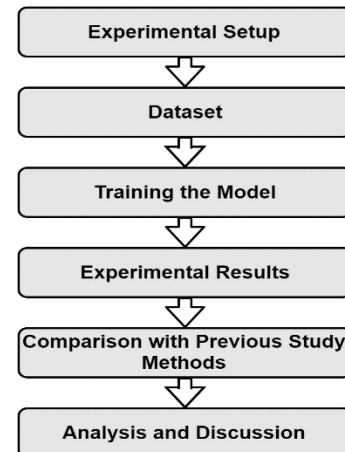


Fig. 4. Experimental framework to validate the proposed method.

A. Experimental Set Up

Proposed method experimented on the environment with a hardware configuration of Intel(R) Core (TM) i7 CPU540 @ 3.07GHz (4 CPUs) ~3.3GHz Processor, 16GB RAM,

Windows 10 64-bit operating system, and MATLAB R2019a. MATLAB provides a very good environment for rapid prototyping and debugging. MATLAB's image processing and machine learning environment provide an efficient set of optimized functions to make the workflow faster. The proposed method requires the parking area image as input and outputs the classified parking areas as green and red.

B. Datasets

The proposed method is experimented using the dataset PKLot [7] for validation. There was a total of 653169 numbers of observations extracted from the actual dataset out of which 72,000 observations are used as training samples. Out of 72,000 observations shown in Table III, 24,000 observations are used from each of the parking areas such as Pontifical Catholic University Paraná - Parking Lot (PUCPR – PKLot), Federal University of Parana - Parking Lot (PKLot - UFPR04) from 4th Floor, Federal University of Parana - Parking Lot (PKLot - UFPR05) from the 5th floor. These 24,000 observations are randomly sampled from the collection of observations for individual parking areas and weather conditions.

TABLE III. NUMBER OF SAMPLES

	Samples per parking area	Number of Training Samples
PUCPR-Sunny	24,000	72,000
PUCPR-Cloudy		
PUCPR-Rainy		
UFPR04-Sunny	24,000	
UFPR04-Cloudy		
UFPR04-Rainy		
UFPR05-Sunny	24,000	

C. Training the Model

72,000 observations are used for training K-nearest neighbor with 10-fold cross-validation for protecting the model from overfitting and the number of neighbors or K used is as 10. After changing different values of K, this research observed that greater than the nearest neighbor value 10, the classification result did not improve and less than k = 10 sometimes leads to false classification. Hence, K=10 is chosen for training the model. Other parameters used for training the KNN model are explained below:

1) *Distance metric*: Euclidean Distance estimated the distance of edges between neighbors as K [27] for the proposed method. Neighbors are m-by-n data vectors where m denotes the number of training samples and n denotes number of features used.

The distance was measured using $d_i(u_s, v_t) = \sqrt{\sum_{i=1}^y (u_{si} - v_{ti})^2}$ where d_i is the distance of the edges between neighbors, u_s and v_t are 1-by-y data vector of the source and destination.

2) *Distance weight*: Weight g_i associated with the training samples are estimated as the squared inverse of the distance, $g_i = \frac{1}{d_i^2}$ and the weighted distance is calculated as

$$d_{g_i}(u_s, v_t) = \sqrt{\sum_{i=1}^y g_i (u_{si} - v_{ti})^2}$$

3) *Break ties*: Smallest value was used to break the tie in case there were an equal number of neighbors with similar values.

4) *Number of nearest neighbors*: Nearest neighbor K=10 used to take decision of new data.

5) *Nearest neighbor search method*: A k-dimensional tree (Kd-Tree) was used to search for the neighbors [28] instead of an exhaustive method to search faster. The Kd-tree method divides a data vector of n-by-k recursively and distributes n points into a binary tree of K-dimension. Hence, the model grows a multi-dimensional Kd-tree using associated weight and Euclidean distance with a bucket size of 60 which is the maximum number of points in the leaf node [29].

D. Evaluation Parameters

The proposed method was validated with the test sets using various metrics, i.e., Confusion Matrix, Accuracy, Area under Curve (AUC), Error, Precision Rate, Recall Rate, Processing Time, and Processing Speed in Frame per Second shown in Fig. 5.

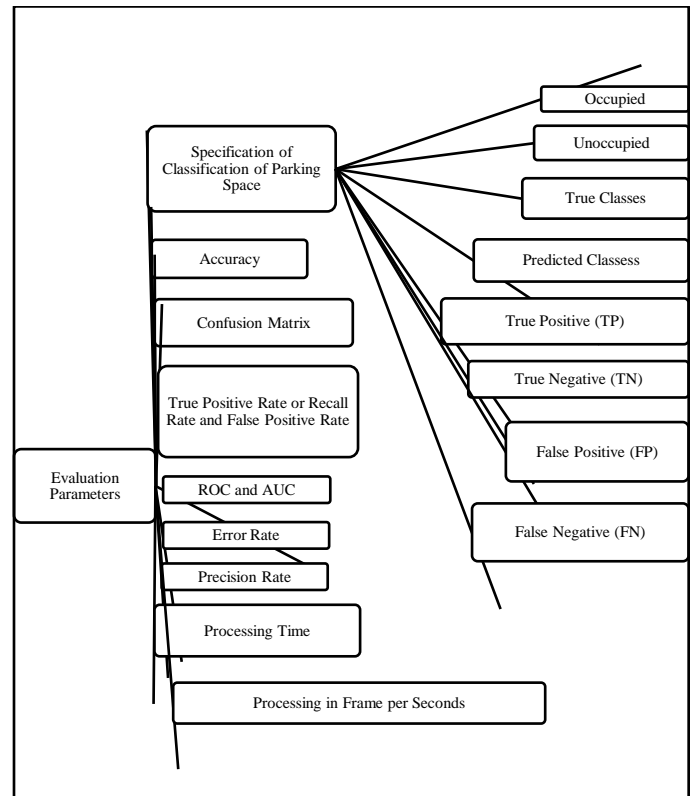


Fig. 5. Performance metrics used for validation.

1) *Specification of the classification of a parking space:* This research used two possible outcomes to denote the status of a parking space, i.e., occupied, and unoccupied. Occupied space refers to when a vehicle is parked in the parking area otherwise denotes unoccupied. These two statuses are considered classes for the prediction model. The true class represents the actual status of parking spaces that are known, and the predicted classes represent the status predicted by the trained model shown in Table VI to Table IX. In addition, looking for unoccupied space is considered a positive interest, and occupied space is considered a negative interest. Classifying a Positive sample as Positive is considered a True Positive (TP), Classifying a Negative sample as Negative is True Negative (TN), Classifying a Negative sample as Positive is False Positive (FP), and Classifying a Positive sample as Negative is False Negative (FN).

2) *Performance metrics:* The proposed method estimated the Confusion matrix which denotes the information about actual status and predicted status of training samples [30]. The performance of the trained model is commonly evaluated using the data of the confusion matrix. True Positive Rate and False Positive Rate were implicated to estimate the Receiver Operating Characteristic curve (ROC) and Area Under Curve (AUC) [31, 32]. The proposed research plotted ROC by placing the FAR on the x-axis and the TPR on the y-axis for several different observations. The values of False Positive Rate (FPR) and True Positive Rate (TPR) range from 0.0 to 1.0. A method with the best prediction skill is represented by the curve that goes from the bottom left corner to the top left corner and then towards the right top corner of the ROC plot [32]. This research also used Precision rate which denotes the rate of the positive prediction in terms of total positive prediction. Processing time was also estimated which denotes the time required to process one frame. In addition, the Processing speed in patches per second is calculated from the number of patches and the processing time taken to process them. In this context, Processing speed in frame per second represents the time to process a frame. Other performance metrics were calculated to validate the proposed method mentioned in Table IV which the corresponding equation used.

E. Experimental Results

The proposed method achieved an accuracy of 99.47% during training, which indicates better performance compared with the state-of-the-art. Table V depicts the confusion matrix of the trained model and Fig. 6 illustrates the AUC of the trained model. The total number of Positive samples (unoccupied spaces) used for the trained model is 36967 and the total number of Negative samples (occupied spaces) was 35033. Out of the total Positive samples, 36743 samples are correctly classified using the trained model, and 224 unoccupied spaces are classified as occupied. Besides, out of the total Negative samples (occupied spaces), 34895 spaces are classified as Negative (occupied), and 138 samples are identified as Positive (unoccupied). Hence, the number of FP and FN is comparatively very small compared to TP and TN

which leads to higher accuracy compared with existing research.

TABLE IV. PERFORMANCE WITH EQUATION

Name of Metrics	Equation
Accuracy [30, 33]	$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$
True Positive Rate or Recall Rate [30, 34]	$True\ Positive\ Rate\ or\ Recall = \frac{TP}{TP + FN}$
False Positive Rate [30, 33, 35]	$False\ Positive\ Rate = \frac{FP}{FP + TN}$
Error Rate [30, 31, 36]	$Error = \frac{FP + FN}{TP + TN + FP + FN}$
Precision Rate [30, 31, 37]	$Precision\ Rate = \frac{TP}{TP + FP}$

TABLE V. CONFUSION MATRIX OF TRAINED KNN MODEL WITH 10-FOLD CROSS-VALIDATION AND K=10

		Predicted Classes	
		Unoccupied	Occupied
True Classes	Unoccupied	36743	224
	Occupied	138	34895

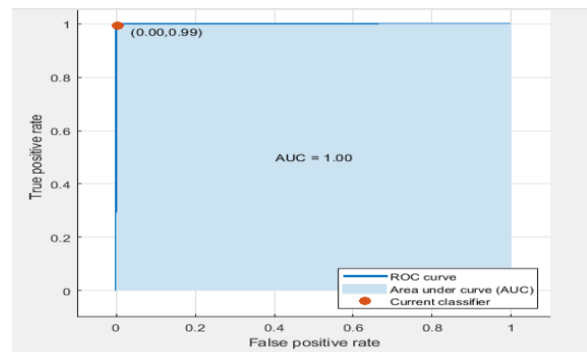


Fig. 6. Area Under Curve (AUC) at training.

Fig. 6 depicts the ROC and AUC curve for the trained model [32]. Here, the curve shown is the ROC curve, the shaded area is AUC which is ≈1 and the point represents the used model. Based on the AUC value, the proposed method performed better for the training samples.

Table VI shows the confusion matrix of tested samples from PUCPR parking area images. 399118 samples were extracted from PUCPR which were used for testing the proposed method, each sample represents a parking space labeled either Unoccupied or Occupied. For the PUCPR images, out of the total samples, 211378 samples are classified as unoccupied, and 2183 samples as occupied out of 213,561 samples that were unoccupied. Again, out of 185,557 samples that were occupied, the proposed method predicted 183,890 samples as occupied and 1,667 samples as unoccupied. There were 88,266 samples from FPR04 parking area images that are used for testing the proposed method to estimate the confusion matrix shown in Table IX. For the FPR04 images, 48,700 samples were classified as unoccupied, and 512 samples as occupied out of a total of 49,212 samples that are unoccupied. Again, out of 39,054 images that were occupied, the proposed method predicted 38,754 samples as occupied and 300

samples as unoccupied. 165785 samples from FPR05 parking area images were extracted using the proposed method to estimate the confusion matrix shown in TABLE X. Proposed method classified 67,361 samples as unoccupied, and 998 samples as occupied out of a total of 68,359 samples that were unoccupied. Out of a total of 97,426 images that were occupied, the proposed method classified 96,648 samples as occupied and 778 samples as unoccupied.

TABLE VI. CONFUSION MATRIX FOR PUCPR, FPR04 AND FPR05

PUCPR		Predicted Classes	
		Unoccupied	Occupied
True Classes	Unoccupied	211378	2183
	Occupied	1667	183890
FPR04			
True Classes	Unoccupied	48700	512
	Occupied	300	38754
FPR05			
True Classes	Unoccupied	67361	998
	Occupied	778	96648

Several observations that were used from different parking areas for validating the model along with the number of prediction speeds in terms of observation/millisecond are shown in Table VII. The prediction speed on average was 23.01 observation (obs)/milliseconds(ms).

TABLE VII. NUMBER OF OBSERVATIONS TESTED AND PREDICTION SPEED

Datasets	Number of Testing Observations	Prediction Speed in o obs/ms
PKLot-PUCPR	Cloudy = 132781	21.23
	Rainy = 83009	
	Sunny = 183329	
PKLot UFPR04	Cloudy = 39392	24.37
	Rainy = 7951	
	Sunny = 40923	
PKLot UFPR05	Cloudy = 56985	23.43
	Rainy = 8929	
	Sunny = 99871	

The accuracy of the proposed method for PUCPR, FPR04, and FPR05 were 99.21%, 99.91%, and 99.29% respectively shown in Table VIII. The accuracy of the three subsets does not fluctuate and it stays above ≈99%. The error rates are 0.79%, 0.9%, and 0.71% which on average stays at around ≈ 0.98%. The precision and recall rate are on average 99.16% and 98.7%. The precision rate represents the rate of the positive prediction in terms of total positive prediction which is 99.16%, and the recall is the rate of positive prediction in terms of total actual positive which is 98.7% for the proposed method.

The number of patches or parking spaces used from different parking spaces is shown in Table IX. PUCPR subset images cover many parking spots in a single image but in the experimentation, 102 parking spots were used from each image of PUCPR, 31 parking spaces from UFPR04, and 44

parking spaces from UFPR05 were used, processing a single image of PUCPR, UFPR04, and UFPR045 with total patches for each required 0.14, 0.15, and 0.17 seconds respectively. Processing speed in patches per second was calculated from the number of patches and the processing time taken to process them. Besides, processing speed in frame per second denotes the time to process a frame shown in Table IX. When the number of patches increased, the processing speed decreased in frame per second because it required more processing time.

TABLE VIII. ACCURACY AND DIFFERENT PERFORMANCE MEASURES

Datasets	Accuracy %	Error %	Precision Rate %	Recall Rate %
PKLot-PUCPR	99.21	0.79	99.27	98.8
PKLot UFPR04	99.91	0.9	99.41	98.7
PKLot UFPR05	99.29	0.71	98.79	98.6

TABLE IX. PROCESSING SPEED IN FRAME PER SECOND

Datasets	Number of Patches in Frame	Processing Time in Seconds	Processing Speed in Patches Per Second	Processing Speed in Frame Per Second
PKLot-PUCPR	102	0.14	44.57	0.39
PKLot UFPR04	31	0.15	41.3	1.41
PKLot UFPR05	44	0.17	47.21	1.09

F. Comparison with Previous Research Performance

The proposed method achieved an average accuracy of 99.47% whereas De Almeida et al. [7] achieved an average accuracy of 91.5% using the PKLot dataset as shown in Table X. In this context, the proposed method by this research achieved an accuracy of 99.47% using the same datasets. Besides, De Almeida et al. [7] used texture-based features from PKLot whereas the proposed method used texture and color descriptor features which caused better performance. CarNet by Nurullayev et al. [8] and AlexNet by Amato et al. [9] received an accuracy of 97.04% and 96.74% respectively which is lower than the proposed method. Deep CNN by Valipour et al. [10] provided an AUC (Area Under Curve) of 0.9994. In this context, AUC achieved by the proposed method is 1 shown in Fig. 6 which indicates proposed method performed better than previous research in terms of accuracy and AUC.

The proposed method requires 0.15 seconds to process a single patch or single parking space shown in Table XI. 1.048 frames was processed in 1 sec which consists of 50 parking spaces considered as a baseline to show the difference with previous research methods. AlexNet by Amato et al. [9] required 0.3 seconds to process a single patch and processed only 0.06 frames in 1 second which was slower than the proposed method. The deep CNN method by Valipour et al. [10] required 0.22 seconds to process a single patch and processed only 0.1 frames per second. So, the proposed method performed better in terms of processing speed than

AlexNet by Amato et al. [9] and the Deep CNN method by Valipour et al. [10].

TABLE X. COMPARISON WITH PREVIOUS RESEARCH

Methods	Accuracy	Error	Precision Rate	Recall Rate
Proposed Method	99.47	0.8	99.15	98.83
PKLot[7]	91.52 (averaged) 99.5 (UFPR05)	0.7 (UFPR05)	99.36 (UFPR05)	98.8 (UFPR05)
CarNet[8]	97.04	-	-	-
AlexNet[9]	96.74	-	-	-
Deep CNN [10]	Accuracy Not provided AUC is (0.9994)	-	-	-

Fig. 7(a), Fig. 7 (b), and Fig. 7(c) are the sample output for PUCPR sunny, cloudy, and rainy weather conditions where

for each image 100 parking spaces are considered for classification. Fig. 7(d), Fig. 7(e), and Fig. 7(f) are the output for FPR04 sunny, cloudy, and rainy weather conditions where for each image 28 parking spaces are used. Fig. 7 (g), Fig. 7(h) and Fig. 7(i) are the output for FPR05 sunny, cloudy, and rainy weather conditions where for each image 40 parking spaces are used for vacant parking space detection. The output result marks the occupied parking space with red rectangles and unoccupied spaces with green rectangles.

TABLE XI. COMPARISON WITH PREVIOUS RESEARCH IN TERMS OF PROCESSING SPEED

Methods	Processing Time in Seconds Per patch	Processing Speed in Frame Per Second (A frame of 50 patches)
Proposed Method	0.15	1.048
PKLot [7]	Not specified	Not specified
CarNet [8]	Not specified	Not specified
AlexNet [9]	0.3	0.06
Deep CNN [10]	0.22	0.1



Fig. 7. Out of detected vacant parking spaces.

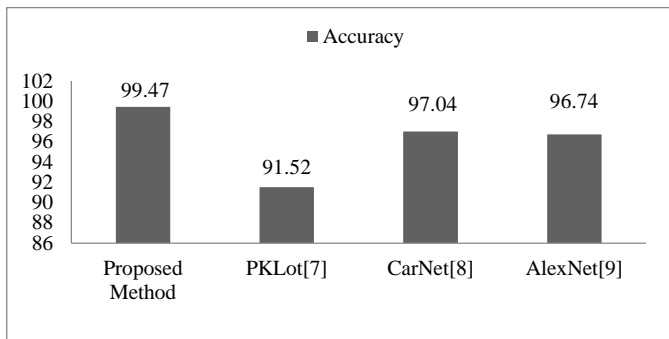
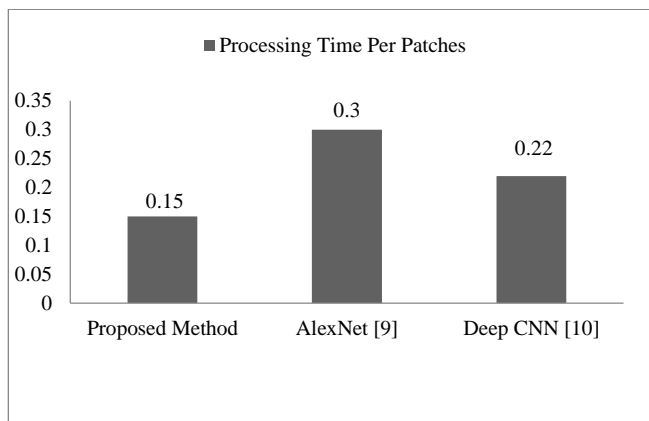
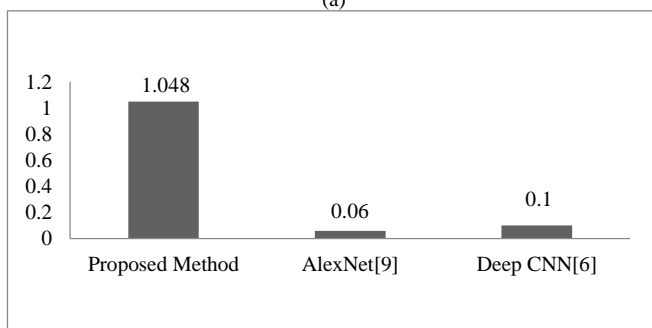


Fig. 8. Accuracy of the proposed method and previous research methods.

The proposed method achieved higher accuracy compared with previous research shown in Fig. 8 due to the usage of features for training, i.e., Contrast, Correlation, Energy, Homogeneity, and Color descriptors such as red, green, and blue channels. In addition, the proposed method also implicates the squared inverse of the distance as a weight to make the classes more separable while training.



(a)



(b)

Fig. 9. (a) And (b) Processing time comparison with previous research.

Proposed methods required less processing time per patch than previous research methods like AlexNet by Amato et al. [9] and the Deep CNN method by Valipour et al. [10] shown in Fig. 9(a). Besides, the proposed method processed more frames per second than AlexNet by Amato et al. [5] and the Deep CNN method by Valipour et al. [10] where each research method considered processing frames of 50 parking spaces shown in Fig. 9(b). The proposed method extracted features from a patch to predict whereas Amato et al. [9] and Valipour et al. [10] used a convolutional neural network and

the whole patch needed to be convolved in each convolution of the neural network which required more time than the proposed method.

The proposed method shows robustness in terms of classifying unseen data. According to the validation results, having enough training data proposed method is expected to be capable to work in newer parking areas. Besides, the proposed method is lightweight and can easily be deployed on the server-side or a single Raspberry Pi 3 or Raspberry Pi 4 can be used to run in real-time implication.

V. CONCLUSION

This research proposes a hybrid method to detect vacant parking spaces using texture-based features and color descriptors. Features are calculated from gray-level co-occurrence matrix and RGB color descriptors to distinguish between occupied and unoccupied spaces. The proposed method illustrated efficiency in terms of accuracy, processing speed, and other performance metrics. Proposed model archived above average accuracy of 99% including validation with unseen data. Experimentation was done on one of the benchmarking datasets named PKLot parking image dataset which contains images of three different parking areas and for each subset images were taken on three different weather conditions at different times of the day. For a desired parking area, the proposed method can be used to provide prior knowledge of the available parking spaces, and vehicle drivers are expected to be able to locate the parking space efficiently. Besides, the proposed method is expected to decrease traffic load and air pollution. Another use case of the proposed method is to detect illegal parking which will be investigated in the future. In addition, an investigation will be done to implicate the proposed method at night and optimize it to work concurrently for both day and night conditions.

ACKNOWLEDGMENT

The authors would like to thank Universiti Kebangsaan Malaysia for providing financial support under the “Geran Universiti Penyelidikan” research grant, GUP-2020-064. The authors also would like to thank West Virginia University Institute of Technology and Colorado State University Pueblo for various support for this research.

REFERENCES

- [1] R. Kaur, R. K. Roul, and S. Batra, "A hybrid deep learning CNN-ELM approach for parking space detection in Smart Cities," *Neural Computing and Applications*, vol. 35, no. 18, pp. 13665-13683, 2023.
- [2] J.-H. Moon and T. K. Ha, "A car parking monitoring system using wireless sensor networks," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 10, pp. 1317-1320, 2013.
- [3] V. Romero Bautista, A. Barreto Flores, S. E. Ayala Raggi, and V. E. Bautista López, "ICM image separation based available parking space detection," *International Journal of Combinatorial Optimization Problems & Informatics*, vol. 14, no. 1, 2023.
- [4] T. Connie, M. K. O. Goh, V. C. Koo, K. T. Murata, and S. Phon-Amnuaisuk, "Improved parking Space recognition via grassmannian deep stacking network with illumination correction," in *International Conference on Computational Intelligence in Information System*, 2021: Springer, pp. 150-159.
- [5] P. R. de Almeida, J. H. Alves, L. S. Oliveira, A. G. Hochuli, J. V. Fröhlich, and R. A. Krauel, "Vehicle Occurrence-based Parking Space Detection," *arXiv preprint arXiv:2306.09940*, 2023.

- [6] L. Li, L. Zhang, X. Li, X. Liu, Y. Shen, and L. Xiong, "Vision-based parking-slot detection: A benchmark and a learning-based approach," in 2017 IEEE International Conference on Multimedia and Expo (ICME), 2017: IEEE, pp. 649-654.
- [7] P. R. De Almeida, L. S. Oliveira, A. S. Britto Jr, E. J. Silva Jr, and A. L. Koerich, "PKLot-A robust dataset for parking lot classification," *Expert Systems with Applications*, vol. 42, no. 11, pp. 4937-4949, 2015.
- [8] S. Nurullayev and S.-W. Lee, "Generalized parking occupancy analysis based on dilated convolutional neural network," *Sensors*, vol. 19, no. 2, p. 277, 2019.
- [9] G. Amato, F. Carrara, F. Falchi, C. Gennaro, C. Meghini, and C. Vairo, "Deep learning for decentralized parking lot occupancy detection," *Expert Systems with Applications*, vol. 72, pp. 327-334, 2017.
- [10] S. Valipour, M. Siam, E. Stroulia, and M. Jagersand, "Parking-stall vacancy indicator system, Based on deep convolutional neural networks," in 2016 IEEE 3rd World Forum on Internet of Things (WF-IoT), 2016: IEEE, pp. 655-660.
- [11] R. M. Nieto, A. Garcia-Martin, A. G. Hauptmann, and J. M. Martinez, "Automatic vacant parking places management system using multicamera vehicle detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 3, pp. 1069-1080, 2018.
- [12] C.-F. Peng, J.-W. Hsieh, S.-W. Leu, and C.-H. Chuang, "Drone-based vacant parking space detection," in 2018 32nd International Conference on Advanced Information Networking and Applications Workshops (WAINA), 2018: IEEE, pp. 618-622.
- [13] A. Regester and V. Paruchuri, "Using computer vision techniques for parking space detection in aerial imagery," in *Advances in Computer Vision: Proceedings of the 2019 Computer Vision Conference (CVC), Volume 2 1*, 2020: Springer, pp. 190-204.
- [14] F. Dornaika, K. Hammoudi, M. Melkemi, and T. Phan, "An efficient pyramid multi-level image descriptor: application to image-based parking lot monitoring," *Signal, Image and Video Processing*, vol. 13, pp. 1611-1617, 2019.
- [15] Q. Li, C. Lin, and Y. Zhao, "Geometric features-based parking slot detection," *Sensors*, vol. 18, no. 9, p. 2821, 2018.
- [16] H. Bura, N. Lin, N. Kumar, S. Malekar, S. Nagaraj, and K. Liu, "An edge based smart parking solution using camera networks and deep learning," in 2018 IEEE International Conference on Cognitive Computing (ICCC), 2018: IEEE, pp. 17-24.
- [17] C.-C. Huang, Y.-S. Tai, and S.-J. Wang, "Vacant parking space detection based on plane-based Bayesian hierarchical framework," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 23, no. 9, pp. 1598-1610, 2013.
- [18] X. Xiang, N. Lv, M. Zhai, and A. El Saddik, "Real-time parking occupancy detection for gas stations based on Haar-AdaBoosting and CNN," *IEEE Sensors Journal*, vol. 17, no. 19, pp. 6360-6367, 2017.
- [19] A. S. Saif, E. D. Wollega, and S. A. Kalevela, "Spatio-Temporal Features based Human Action Recognition using Convolutional Long Short-Term Deep Neural Network," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 5, 2023.
- [20] L. Baroffio, L. Bondi, M. Cesana, A. E. Redondi, and M. Tagliasacchi, "A visual sensor network for parking lot occupancy detection in smart cities," in 2015 IEEE 2nd World Forum on Internet of Things (WF-IoT), 2015: IEEE, pp. 745-750.
- [21] 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.
- [22] H. T. Vu and C.-C. Huang, "Parking space status inference upon a deep CNN and multi-task contrastive network with spatial transform," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 1194-1208, 2018.
- [23] A. S. Saif and Z. R. Mahayuddin, "Stereo Vision Based Localization of Handheld Controller in Virtual Reality for 3D Painting Using Inertial System," *Journal of image and graphics*, vol. 11, no. 2, pp. 127-131, 2023.
- [24] G. Amato, F. Carrara, F. Falchi, C. Gennaro, and C. Vairo, "Car parking occupancy detection using smart camera networks and deep learning," in 2016 IEEE symposium on computers and communication (ISCC), 2016: IEEE, pp. 1212-1217.
- [25] A. S. Saif and Z. R. Mahayuddin, "Augmented Reality-Based 3D Human Hands Tracking From Monocular True Images Using Convolutional Neural Network," in *Handbook of Research on Artificial Intelligence and Knowledge Management in Asia's Digital Economy: IGI Global*, 2023, pp. 129-137.
- [26] A. S. Saif and Z. R. Mahayuddin, "Robust Analysis of Motor Imagery From Brain Signals for a BCI-Controlled Virtual Reality System to Aid Paralysis Patients," in *Handbook of Research on Artificial Intelligence and Knowledge Management in Asia's Digital Economy: IGI Global*, 2023, pp. 119-128.
- [27] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *Journal of machine learning research*, vol. 10, no. 2, 2009.
- [28] C. Silpa-Anan and R. Hartley, "Optimised KD-trees for fast image descriptor matching," in 2008 IEEE Conference on Computer Vision and Pattern Recognition, 2008: IEEE, pp. 1-8.
- [29] A. Saif and Z. R. Mahayuddin, "Crowd density estimation from autonomous drones using deep learning: challenges and applications," *Journal of Engineering and Science Research*, 2021.
- [30] V. Labatut and H. Cherifi, "Accuracy measures for the comparison of classifiers," *arXiv preprint arXiv:1207.3790*, 2012.
- [31] A. S. Saif and Z. R. Mahayuddin, "Vision based 3D Object Detection using Deep Learning: Methods with Challenges and Applications towards Future Directions," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 11, 2022.
- [32] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern recognition*, vol. 30, no. 7, pp. 1145-1159, 1997.
- [33] M. A. S. Khan, M. J. B. Showmik, T. Ahmed, and A. S. Saif, "A Constructive Review on Pedestrian Action Detection, Recognition and Prediction," in *Proceedings of the 2nd International Conference on Computing Advancements*, 2022, pp. 367-376.
- [34] F. Rasheed, Y. Saleem, K. L. A. Yau, Y. W. Chong, and S. L. Keoh, "The Role of Deep Learning in Parking Space Identification and Prediction Systems," *Computers, Materials and Continua*, vol. 75, no. 1, pp. 761-784, 2023.
- [35] M. M. Abdellatif, N. H. Elshabasy, A. E. Elashmawy, and M. AbdelRaheem, "A low cost IoT-based Arabic license plate recognition model for smart parking systems," *Ain Shams Engineering Journal*, vol. 14, no. 6, p. 102178, 2023.
- [36] G. Satyanath, J. K. Sahoo, and R. K. Roul, "Smart parking space detection under hazy conditions using convolutional neural networks: a novel approach," *Multimedia Tools and Applications*, vol. 82, no. 10, pp. 15415-15438, 2023.
- [37] R. T. Maharshi, D. Nagajyothi, P. Thrishul, and P. Reethika, "A System for Detecting Automated Parking Slots Using Deep Learning," in 2023 International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE), 2023: IEEE, pp. 1-6.