

Parking detection system using background subtraction and HSV color segmentation

Awang Hendrianto Pratomo, Wilis Kaswidjanti, Alek Setiyo Nugroho, Shoffan Saifullah

Department of Informatics, Universitas Pembangunan Nasional Veteran Yogyakarta, Indonesia

Article Info

Article history:

Received Aug 31, 2021

Revised Oct 31, 2021

Accepted Nov 2, 2021

Keywords:

Background subtraction

HSV

Image segmentation

Parking detection system

Smart parking

ABSTRACT

Manual system vehicle parking makes finding vacant parking lots difficult, so it has to check directly to the vacant space. If many people do parking, then the time needed for it is very much or requires many people to handle it. This research develops a real-time parking system to detect parking. The system is designed using the HSV color segmentation method in determining the background image. In addition, the detection process uses the background subtraction method. Applying these two methods requires image preprocessing using several methods such as grayscaling, blurring (low-pass filter). In addition, it is followed by a thresholding and filtering process to get the best image in the detection process. In the process, there is a determination of the ROI to determine the focus area of the object identified as empty parking. The parking detection process produces the best average accuracy of 95.76%. The minimum threshold value of 255 pixels is 0.4. This value is the best value from 33 test data in several criteria, such as the time of capture, composition and color of the vehicle, the shape of the shadow of the object's environment, and the intensity of light. This parking detection system can be implemented in real-time to determine the position of an empty place.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Wilis Kaswidjanti

Department of Informatics

Universitas Pembangunan Nasional Veteran Yogyakarta

Jl Babarsari 2 Yogyakarta, 55281, Indonesia

Email: wilisk@upnyk.ac.id

1. INTRODUCTION

Parking space is a facility that must be fulfilled [1] both in business, shopping, public places, and institutions. Parking facilities play an essential role in supporting the progress of the place [2]. Thus, a well-organized parking system makes it comfortable for motorists. Currently, many conventional parking systems cause parking users to find it difficult to find empty parking spaces. The reason is due to the lack of information on the parking lot. The extent of the parking area and the number of parked vehicles cause parking users to have to circle the parking lot first to find an empty parking space. Thus, this system is inefficient because it can take more time and cost [3]. On these problems, the research develops a real-time parking detection system which will be developed into a smart parking system [4].

The smart parking system was developed for parking management and monitoring. The current development uses a sensor-based and image-based approach [5]. Sensor-based monitoring uses infrared to detect the availability of parking lots. This sensor requires a high cost to implement. In addition, the sensor can only detect one parking lot, so it requires many sensors to detect several parking lots simultaneously. Compared with image processing with computer vision, it can detect objects with a broader range than infrared. This concept uses the camera as an input device to detect the object.

Based on the concept of computer vision in object detection, this method aims to visualize and analyze images acquired from cameras using digital image processing techniques [6]. One of the acquisition processes is using a closed-circuit television (CCTV) camera, which produces digital images in image processing. The process of detecting the availability of parking lots with CCTV can be done with a marker [7]. Markers are placed in parking lots, where the camera can detect signs that parking spaces are considered vacant. The parking lot is considered occupied if the camera does not detect a marker. In the detection process, bright enough light rays can bounce off the camera to detect the motif. This parking detection process has been carried out using the canny detection method [8] and dilation [9]. Both methods are implemented to determine the vacant parking lot [10]. However, the Canny edge detection method relies on the parking lot boundary line, so the camera must be directed to get the image at the right angle. Meanwhile, the dilation method still has errors in detecting parking spaces, especially when the parking lot is between two parked cars.

Dependence detection on specific objects such as markers and parking dividing lines can be solved using the background subtraction method [11]. The background subtraction method compares two camera-captured images to detect and track vehicles in the parking area [12]. The background subtraction method is susceptible to changes in light [13], [14]. So in the process, the method uses the concept of reduction by sampling background images under certain conditions, such as morning, afternoon, evening, and night. The background image sampling uses the image energy variable to get a background image that matches the foreground. However, under certain conditions, the detection results are wrong. The detection error is caused by the shadow of the object's environment [15], which is detected as a vehicle. This condition causes the accuracy to be less than optimal. The shadow effect can be corrected by recognizing the shadow condition of the object from the foreground image [16], which selects the image background to increase accuracy.

Color image segmentation [17] is our proposal to improve image quality from shadow effects. This method uses the hue, saturation, value (HSV) color of the shadow. HSV can segment images based on color by utilizing the upper and lower values of the HSV value so that objects and backgrounds are separated and can segment noisy color images. HSV accuracy is also better in segmenting color images than red, green, dan blue (RGB), HSL, and L*A*B [18]. This study aims to provide a solution in real-time parking detection using a combination of background subtraction HSV segmentation methods to determine the background image.

In this article, there are four main sections. The first section describes the background, novelty of the study, and objectives. Next, it presents the methods used in image processing-based parking detection in the second part. The third part conveys the results and discussion of the process and system testing that has been designed. Finally, the fifth section presents conclusions on the results obtained and their tests.

2. RESEARCH METHOD

This study uses quantitative methods by testing the effect of a variable in the study. In general, the stages are shown in Figure 1. This study uses the concept of image processing to detect empty parking lots. The methods used include searching for image background, preprocessing, background subtraction, and filtering.

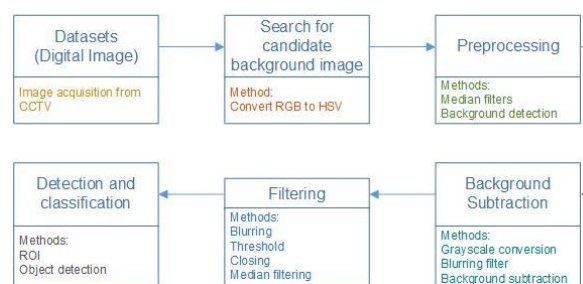


Figure 1. Steps of parking detection system on image processing

2.1. Datasets

The datasets of this study used images acquired by CCTV from the parking lot of the Communication and Information Office of Gunungkidul Regency. The image has a background that contains partial or full shadows. The shadow part of the image constantly changes due to changes in the direction of

light hitting the object (e.g., buildings, trees, or other objects). In comparison, the full shadow has a similar shape to the shadow. Based on direct observation, the shadow is due to the presence of sunlight blocked by clouds. Therefore, we divide the images between 8.00-16.00 and are acquired once every 15 minutes. It aims so that the image's shape with a partial shadow does not have a shadow shape. However, this does not apply to images with full shadows (shadow shapes resemble objects at any given time). The data were tested using foreground image data with different capture times, vehicle composition and color, the shape of the object's environmental shadow, and different light intensities.

2.2. Search for candidate image background

The background image candidate search focuses on image processing to determine the candidate background image dataset used based on the average number of pixels of 255 per image. The stages of searching for candidate background images as:

- Background image is an acquired image that contains two types of data, namely images with partial and full shadows. The image data used are 20 image samples at each time frame.
- The image is converted from an RGB image to an HSV image, with three elements: hue representing color, saturation representing color intensity level, and value representing brightness level. The HSV method also has an excellent ability to segment color images that have noise. HSV's accuracy is better in segmenting images based on color than RGB, HSL, and L*A*B [18]. HSV calculation can be done using (1).

$$C_{max} = \max(R, G, B), C_{min} = \min(R, G, B)$$

$$C = C_{max} - C_{min}$$
(1)

After the min and max values for each color are obtained, the values for each HSV element can be calculated using (2). This calculation uses the reference of (1).

$$H = \begin{cases} 60 \left(\frac{G - B}{C} \right), & \text{if } C_{max} = R \\ 120 + 60 \left(\frac{B - R}{C} \right), & \text{if } C_{max} = G \\ 240 + 60 \left(\frac{R - G}{C} \right), & \text{if } C_{max} = B \end{cases}$$
(2)

$$S = \begin{cases} 0, & C_{max} = 0 \\ \frac{C}{C_{max}}, & C_{max} \neq 0 \end{cases}$$

$$V = C_{max}$$

- HSV can segment color images by utilizing the upper and lower values of the HSV value to separate objects from the background. This value can be calculated by trial and error and produces a lower value (0, 0, 0) and an upper value (179, 255, 85).
- Noise reduction uses the median filter method to reduce noise better than the same linear smoothing model [19]. The following is the equation for the median filter equation.

$$x = \frac{n+1}{2}$$
(3)

- After the median filter process, the next step is to find the number of pixels of 255 (white color) per existing frame.
- Find an average of 255 pixels for each image frame.

$$Average = \frac{\text{total of 255 pixel every frame}}{\text{total of frame}}$$
(4)

- The average result becomes a reference for finding candidate background images.

2.3. Preprocessing

Preprocessing is used to get the selected background image from the existing candidate background image with the shadow condition closest to the foreground image (real-time) [20]. In addition, this process aims to minimize shadow noise from the object's environment. The preprocessing stages carried out in this study include as:

- The candidate background image and foreground image (real-time) are converted to the HSV color space, then segmented by shadow color and filter median as in the previous step.
- The stage of determining the background image uses the slightest 255-pixel difference between the candidate background and foreground images.
- The candidate background image that has the slightest 255-pixel difference becomes the selected background image.

2.4. Background subtraction

The selected background image becomes a real-time reference for performing background subtraction between the background and foreground images. The stages of background subtraction include as:

- Grayscale is a digital image that has only one channel value per pixel. This value indicates the level of color intensity ranging from black, gray, and white. This image in each pixel has a color from white to black, where each pixel is represented by 8 bits. The grayscale process can be done using (5) [21], [22].

$$Y_{(x,y)} = (0,229 \times R) + (0,587 \times G) + (0,114 \times B) \quad (5)$$

- Blurring is a low-pass filter that takes an image with a smooth, intense gradation. High-intensity differences are reduced or removed to reduce noise in an image. The blurring process itself aims to change the intensity of the noise image to gray so that when the thresholding process is carried out, the noise image will be reduced. The gaussian distribution equation can be seen in the following equation [23].

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}} = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{x^2}{2\sigma^2}} \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{y^2}{2\sigma^2}} = G(x)G(y) \quad (6)$$

- Background subtraction is one method of the segmentation process that separates moving objects (foreground) and background objects (background) [24]. Background subtraction has a vital role in computer vision, one of which is monitoring systems. The role played in background subtraction is knowing or distinguishing between the background and objects in an image. The following is the background subtraction (7):

$$R_k(x, y) = |f_k(x, y) - B_k(x, y)|$$

$$D_k(x, y) = \begin{cases} 255, & \text{background } R_k(x, y) > T \\ 0, & \text{background } R_k(x, y) \leq T \end{cases} \quad (7)$$

Background subtraction on the foreground image (real-time image) and the selected background image, each of which has been blurred in the previous step. The purpose of segmentation [25] using background subtraction is to obtain objects from the resulting difference between the foreground and the selected background images. The results of background subtraction can be seen in Figure 2.

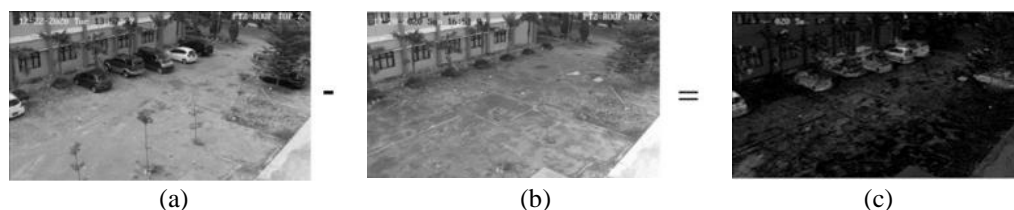


Figure 2. The results of background subtraction with, (a) foreground image, (b) background image, (c) difference

- d. Thresholding is used to filter the noise contained in an image. Thresholding is the process of converting a grayscale image into a binary or black and white image so that it can be seen between the foreground and background in an image, separating the pixel values according to a predetermined threshold. Thresholding can be written as 8.

$$G_{(x,y)} = \begin{cases} \text{is 255 if } F_{(x,y)} > T \\ \text{is 0 if } F_{(x,y)} \leq T \end{cases} \quad (8)$$

The following is the result of the background subtraction stage after the threshold is carried out, as shown in Figure 3.



Figure 3. Sample of the results image of thresholding

2.5. Filtering

Filtering is a process to take or remove specific frequencies from an image [26]. At this stage, it is done to reduce or minimize the existing noise. It was starting from doing blurring and thresholding again. It was then continued with closing morphology and median filter.

- The blurring is done again because the image produced in the previous stage is still too rough.
- Threshold is done again to emphasize the results of blurring that is done.
- The results of the background subtraction method show that there is still noise in the image results, then to remove the noise, the researcher uses morphological operations with the opening method. The closing morphology is the opposite of the opening morphology. In the closing morphology, dilation is carried out first and then followed by erosion. Dilation enlarges the binary image by adding layers around the object, while erosion is the opposite, reducing or eroding the object's edges.
- In the image of the closing morphological results, there is still noise in the form of object spots in the image. Therefore, to reduce the noise, the researchers used the median filter method. The median filter method takes a specific area of the image according to predetermined kernel size. The filtered image can be seen in Figure 4.



Figure 4. Filtering result on the parking area

2.6. Detection and classification

The detection process is a process to get an object in the form of 255 pixels. The detection stages as:

- Region of interest (ROI) works in coding differently in some regions of the digital image. The more critical image area will have a better image quality than the surrounding area. Thus, ROI can be used to limit the area of a vehicle. Therefore, a ROI formed a focus area for detecting pixel 255 in each parking slot available in the parking lot. The following is an example of ROI, as shown in Figure 5.
- The number of 255 pixels contained in each ROI is the result of the resulting detection.

Classification is carried out based on the minimum percentage of objects in the form of white pixels (255) to the total pixels in each area. If an area has a pixel percentage of 255 less than the minimum percentage limit, then the parking lot is considered empty. At the same time, an area with an object percentage is more than the minimum limit, and then the parking lot is considered occupied or filled.

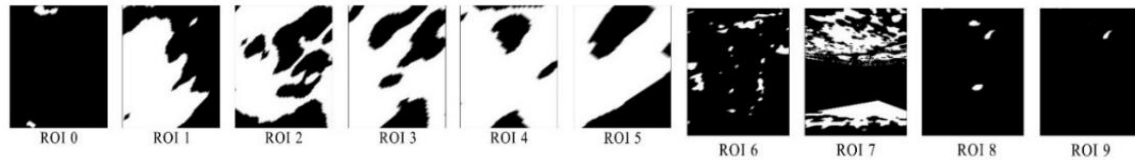


Figure 5. ROI

3. RESULTS AND DISCUSSION

The result of this research is the detection of empty parking spaces. The final result was tested to determine the level of accuracy resulting from the background subtraction method and the hue saturation value for parking detection. Accuracy testing is done by observing the detection of parking lots using the method used with the actual data obtained. The test data consisted of 33 foreground images obtained from random CCTV camera captures during sunny conditions from 8.00 to 16.00. The foreground image has different characteristics from the time of image capture, the shape of the shadow contained in the image, the intensity of light, and the composition and color of the vehicle parked in it (number and type of car).

The level of accuracy is calculated using the confusion matrix, namely by comparing the classification results from parking detection results from the introduction of the system with the actual parking slots. The classification uses a different minimum threshold value of 255 pixels to get the best accuracy value. The minimum threshold value of 255 pixels used to classify the test data used is between 0 to 1, with a difference of 0.1.

The accuracy test with the confusion matrix has four terms to represent the results of the classification process, namely true positive (TP), true negative (TN), false positive (FP), and false negative (FN) [27]. The TP value is obtained if the number of parking slots filled with vehicles is detected correctly by the system, and the TN value is obtained if the number of available (empty) parking slots is detected correctly by the system. The FP value is obtained if the number of parking slots filled with vehicles is detected incorrectly by the system. The FN value is obtained if the number of parking slots filled with vehicles is detected incorrectly by the system. An available (empty) parking slot was detected incorrectly by the system. An example of testing on one of the test data is shown in Figure 6 (a). The third test data has contained in 12 parking slot images. Figure 6 is one of the test data used for this study. In the test image, the coordinates of each existing parking slot are determined first. The coordinates of the parking slots are used as a ROI after the image is processed. The image processing results are shown in Figure 6 (b), which is not yet clear for the desired slot. So, this process needs to be transformed to get more apparent results.

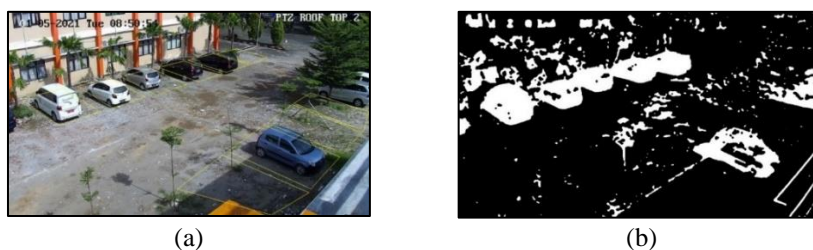


Figure 6. These figures are, (a) sample of the third testing data, (b) the result of image processing on the third testing data

The transformation process uses a region of interest concept with a perspective transformation at a predetermined coordinate point. The height of the perspective image in this study is 200 px, and the width is 120 px. The establishment of a region of interest aims to focus the detection process on a certain point. The identification of ROI images based on these images is produced in Figure 7, starting from ROI 0 to ROI 9. This ROI shows that each part is a reference for the empty parking detection process. ROI that detects 255

pixels will be classified as a parking lot. The processed ROI has different threshold values and classifications based on the calculation, as shown in Table 1. In Table 1, an example of the implementation of ROI from the third image data processed with ten total ROI and the variation of the threshold value between 0 and 1 with a difference of 0.1 shows different results.

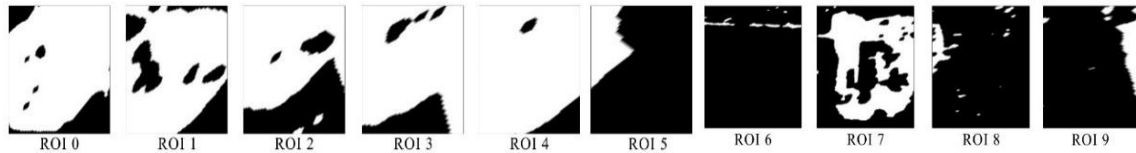


Figure 7. Results of the region of interest

Table 1. The results of classification on the third testing data

ROI	Pixel 255	Pixel ROI (Area)	Value	Threshold											
				0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1		
ROI 0	27096	32000	0.84675	x	x	x	x	x	x	x	x	x	x	✓	✓
ROI 1	24860	32000	0.776875	x	x	x	x	x	x	x	x	✓	✓	✓	✓
ROI 2	20781	32000	0.64940625	x	x	x	x	x	x	✓	✓	✓	✓	✓	✓
ROI 3	26730	32000	0.8353125	x	x	x	x	x	x	x	x	✓	✓	✓	✓
ROI 4	29555	32000	0.92359375	x	x	x	x	x	x	x	x	x	x	✓	✓
ROI 5	5669	32000	0.17715625	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ROI 6	714	32000	0.0223125	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ROI 7	16234	32000	0.5073125	x	x	x	x	x	✓	✓	✓	✓	✓	✓	✓
ROI 8	1883	32000	0.05884375	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
ROI 9	3800	32000	0.11875	x	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

Note: =parking slots filled; =parking slots available

The classification process was tested using the confusion matrix method. An example of accuracy testing is shown in Figure 8. Figure 8 has three kinds of detection because it uses different thresholds. The thresholds used are 0.1, 0.4, and 0.8. The the accuracy calculation of this experiment can be seen in Table 2.

Based on the tests in Table 2, the third test data with a threshold of 0.1 has an accuracy of 80%. This first test has a value of FN=2 because there is an empty parking slot but identifies the presence of a vehicle. It can be seen from the color of the red box, which identifies that the parking lot has been filled. Two empty slots were boxed in red from the experiment, caused by the light covering/blocking the parking lot. Thud, the following experiment is to increase the threshold value to 0.4; when this threshold is implemented, the accuracy increases to 100%, where the condition of all parking slots is classified correctly.



Figure 8. Detection results on the third testing data using threshold, (a) 0.1, (b) 0.4, (c) 0.8

Table 2. The accuracy of testing data based on Figure 8

Figure	Threshold				Accuracy (%)
	TP	TN	FP	FN	
9.(a)	6	2	0	2	80
9.(b)	6	4	0	0	100
9.(c)	3	4	3	0	70

In addition, the third test is to use a more significant threshold of 0.8. This experiment yielded an accuracy of 70%. The test has a value of FP=3. This situation is because the parking slot is filled with

vehicles but is considered empty by the system. The slot that is considered empty is probably because the detection of the image used shows that the object image is close to the background image. At the end of the accuracy testing process, each threshold value in the overall test data is summed and averaged. These results are used to obtain the maximum accuracy value. The average accuracy results for each test with various threshold values are shown in Figure 9.

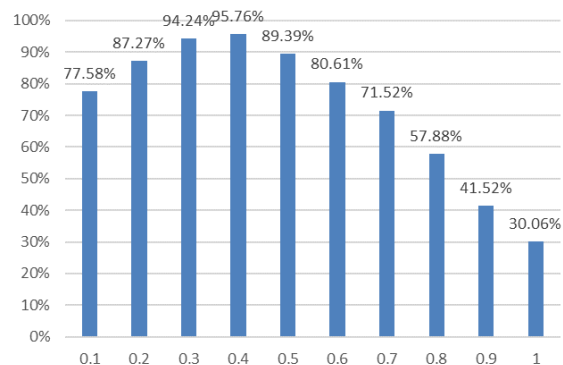


Figure 9. Detection results on the third testing data using threshold=0.1

Based on Figure 9, the average accuracy of all tested data is at a different minimum threshold value of 255 pixels. The highest accuracy result is 95.76%, with a minimum threshold value of 0.4. At the same time, the lowest accuracy is 30.06%, with a minimum threshold value of 1. The detection results can provide the best accuracy for classifying empty parking slots.

4. CONCLUSION

The background subtraction method can be implemented in determining the image background. The addition of the HSV segmentation method in determining the background provides the best detection results with an average accuracy of 95.76%. This testing process is based on a minimum threshold value of 0.4 at 255 pixels. This threshold has the best resistance from 33 test data based on several factors: the time of capture, vehicle composition and color, the shape of the object's environmental shadow, and different light intensities. However, the detection process is still very dependent on the position and angle of the CCTV camera to the parking lot. This condition results in the possibility of errors still occurring at certain parking positions. This error occurs because the system detects vehicles in the parking lot next to the detected object is considered an incoming object, and objects other than vehicles contained in the parking lot are still considered occupied parking lots. In the future, to improve this, it is necessary to apply a machine vision method that detects objects according to the type of vehicle. It also needed to set the equipment of image/video acquisition to obtain a specific object.

REFERENCES

- [1] J. Ortega, J. Hamadneh, D. Esztergár-Kiss and J. Tóth, "Simulation of the Daily Activity Plans of Travelers Using the Park-and-Ride System and Autonomous Vehicles: Work and Shopping Trip Purposes," *Applied Sciences*, vol. 10, no. 8, p. 2912, April 2020, doi: 10.3390/app10082912.
- [2] A. Sharma and R. B. Battula, "Smart City: Evolution and Fundamental Concepts," in *Security and Organization within IoT and Smart Cities*, CRC Press, 2020, pp. 19-40.
- [3] K. I. M. Ata, A. C. Soh, A. J. Ishak, H. Jaafar and N. A. Khairuddin, "Smart Indoor Parking System Based on Dijkstra's Algorithm," *International Journal of Electrical Engineering and Applied Sciences*, vol. 2, no. 1, pp. 13-20, 2019.
- [4] G. Ali *et al.*, "IoT Based Smart Parking System Using Deep Long Short Memory Network," *Electronics*, vol. 9, no. 10, p. 1696, October 2020, doi: 10.3390/electronics9101696.
- [5] P. Jmsmg and H. Wickramarachchi, "CCTV Based Parking Occupancy Tracker Using Computer Vision," in *Proceedings of the 11 Symposium on Applied Science, Business & Industrial Research*, December 2019, pp. 18-24, [Online]. Available: https://www.researchgate.net/publication/338001800_CCTV_Based_Parking_Occupancy_Tracker_Using_Computer_Vision.
- [6] S. Shih and W. Tsai, "A Convenient Vision-Based System for Automatic Detection of Parking Spaces in Indoor Parking Lots Using Wide-Angle Cameras," in *IEEE Transactions on Vehicular Technology*, vol. 63, no. 6, pp. 2521-

- 2532, July 2014, doi: 10.1109/TVT.2013.2297331.
- [7] K. B. Dsouza, S. Mohammed and Y. Hussain, "Smart parking-An integrated solution for an urban setting," *2017 2nd International Conference for Convergence in Technology (I2CT)*, 2017, pp. 174-177, doi: 10.1109/I2CT.2017.8226115..
- [8] A. Kanáliková and E. Bubeníková, "Parking system with image processing," *2019 IEEE 17th World Symposium on Applied Machine Intelligence and Informatics (SAMi)*, 2019, pp. 281-286, doi: 10.1109/SAMI.2019.8782760..
- [9] M. Noor and A. Shrivastava, "Automatic Parking Slot Occupancy Detection using Laplacian Operator and Morphological Kernel Dilation," *2021 10th IEEE International Conference on Communication Systems and Network Technologies (CSNT)*, 2021, pp. 825-831, doi: 10.1109/CSNT51715.2021.9509620.
- [10] B. Kommey, E. O. Addo and A. S. Agbemenu, "A Smart Image Processing-based System for Parking Space Vacancy Management," *International Journal of Computer Applications*, vol. 182, no. 5, pp. 1-6, July 2018, doi: 10.5120/ijca2018917540.
- [11] B. Garcia-Garcia, T. Bouwmans and A. J. R. Silva, "Background Subtraction in Real Applications: Challenges, Current Models and Future Directions," vol. 35, p. 100204, February 2020, doi: 10.1016/j.cosrev.2019.100204. [Online]. Available: <http://arxiv.org/abs/1901.03577>.
- [12] C. G. Postigo, J. Torres and J. M. Menéndez, "Vacant parking area estimation through background subtraction and transience map analysis," *IET Intelligent Transport Systems*, vol. 9, no. 9, pp. 835-841, November 2015, doi: 10.1049/iet-its.2014.0090.
- [13] X. Lu, C. Xu, L. Wang and L. Teng, "Improved background subtraction method for detecting moving objects based on GMM," *IEEJ Transactions on Electrical and Electronic Engineering*, vol. 13, no. 11, pp. 1540-1550, November 2018, doi: 10.1002/tee.22718.
- [14] Y. Zhang, W. Zheng, K. Leng and H. Li, "Background Subtraction Using an Adaptive Local Median Texture Feature in Illumination Changes Urban Traffic Scenes," *IEEE Access*, vol. 8, pp. 130367-130378, 2020, doi: 10.1109/ACCESS.2020.3009104.
- [15] M. Wu, R. Chen and Y. Tong, "Shadow Elimination Algorithm Using Color and Texture Features," *Computational intelligence and neuroscience*, pp. 1-10, January 2020, doi: 10.1155/2020/2075781.
- [16] S. M. Roy and A. Ghosh, "Foreground Segmentation Using Adaptive 3 Phase Background Model," in *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 6, pp. 2287-2296, June 2020, doi: 10.1109/TITS.2019.2915568.
- [17] S. Saifullah, "K-means Segmentation Based-on Lab Color Space for Embryo Egg Detection," *arXiv Prepr. arXiv2103.02288*, March 2021. [Online]. Available: <http://arxiv.org/abs/2103.02288>.
- [18] J. Pardede, M. G. Husada, A. N. Hermana and S. A. Rumapea, "Fruit Ripeness Based on RGB, HSV, HSL, L*a*b* Color Feature Using SVM," in *2019 International Conference of Computer Science and Information Technology (ICoSNiKOM)*, 2019, pp. 1-5, doi: 10.1109/ICoSNiKOM48755.2019.9111486.
- [19] S. Suhas and C. R. Venugopal, "MRI image preprocessing and noise removal technique using linear and nonlinear filters," *2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT)*, 2017, pp. 1-4, doi: 10.1109/ICEECCOT.2017.8284595..
- [20] P. Zheng, D. Qin, B. Han, L. Ma and T. M. Berhane, "Research on Feature Extraction Method of Indoor Visual Positioning Image Based on Area Division of Foreground and Background," *ISPRS International Journal of Geo-Information*, vol. 10, no. 6, p. 402, Jun. 2021, doi: 10.3390/ijgi10060402.
- [21] S. Saifullah, "Segmentation for embryonated Egg Images Detection using the K-Means Algorithm in Image Processing," *2020 Fifth International Conference on Informatics and Computing (ICIC)*, 2020, pp. 1-7, doi: 10.1109/ICIC50835.2020.9288648.
- [22] Sunardi, A. Yudhana and S. Saifullah, "Identity analysis of egg based on digital and thermal imaging: Image processing and counting object concept," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 7, no. 1, pp. 200-208, February 2017, doi: 10.11591/ijece.v7i1.pp200-208.
- [23] P. Singhal, A. Verma and A. Garg, "A study in finding effectiveness of Gaussian blur filter over bilateral filter in natural scenes for graph based image segmentation," *2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS)*, 2017, pp. 1-6, doi: 10.1109/ICACCS.2017.8014612.
- [24] H. Wang and L. Shi, "Foreground model for background subtraction with blind updating," *2016 IEEE International Conference on Signal and Image Processing (ICSIP)*, 2016, pp. 74-78, doi: 10.1109/SIPROCESS.2016.7888227.
- [25] A. Yudhana, Sunardi and S. Saifullah, "Segmentation comparing eggs watermarking image and original image," *Bulletin of Electrical Engineering and Informatics*, vol. 6, no. 1, pp. 47-53, March 2017, doi: 10.11591/eei.v6i1.595.
- [26] K. Han, Z. Wang and Z. Chen, "Fingerprint Image Enhancement Method based on Adaptive Median Filter," *2018 24th Asia-Pacific Conference on Communications (APCC)*, 2018, pp. 40-44, doi: 10.1109/APCC.2018.8633498..
- [27] S. Saifullah, Y. Fauziah and A. S. Aribowo, "Comparison of Machine Learning for Sentiment Analysis in Detecting Anxiety Based on Social Media Data," *arXiv preprint arXiv:2101.06353*, January 2021. [Online]. Available: <http://arxiv.org/abs/2101.06353>.