Received: August 7, 2021. Revised: September 13, 2021.

International Journal of
Intelligent Engineering & Systems

http://www.inass.org/

Parking Lots Detection in Static Image Using Support Vector Machine Based on Genetic Algorithm

Yuslena Sari[1]* **Hargokendar Suhud[2]** **Andreyan Rizky Baskara[1]**
Ricardus Anggi Pramunendar[3] **Iphan F. Radam[1]**

[1]*Universitas Lambung Mangkurat, Banjarmasin 70123, Indonesia*
[2]*STIKES Karya Husada, Semarang, Indonesia*
[3]*Universitas Dian Nuswantoro, Semarang, Indonesia*
* Corresponding author's Email: yuzlena@ulm.ac.id

Abstract: The increasing ownership of four-wheeled vehicles creates a new problem: difficulty finding available parking spaces in significant places. Issues citizens often experience these in the city centre who use their cars for transportation. This causes the detection of parking lots to attract the attention of researchers. Unfortunately, this system fails when the vehicle occupies more than one place or when the parking lot has a different parking space. In this study, an automation method is proposed that utilises a combination of gray level co-occurrence matrix (GLCM) and support vector machine (SVM) methods with genetic algorithm (GA) optimisation techniques or called SVMGA. The results showed that the SVMGA could provide an accuracy performance of 96.99% for training data and 94.36% for test data. The results of this study are expected to help the queue of people to find a parking space and reduce the density of lines in the parking lot.

Keywords: Parking detection, Gray level co-occurrence matrix, Support vector machine, Genetic algorithm.

1. Introduction

With the increasing number of cars or, more precisely, drivers looking for parking spaces in shopping centre parking areas, fast and accurate information regarding empty parking spaces are increasingly needed. Competition between drivers in finding a parking space is inevitable. The time wasted looking for an open parking space in the parking area can trigger the driver's stress level to increase and increase the workload experienced by the driver. Drivers need fast and accurate information in finding an empty parking space in the parking area, so it does not waste driver's time and maintaining the level of competition between drivers in finding a parking space. Therefore the driver's stress level is not increased.

Current technological developments can provide solutions to these problems. One of them is digital image processing, known as image processing. Digital imaging techniques can give the driver the information needed to find an empty parking space in the parking area quickly and accurately without being constrained by conditions (dark) or weather (rain).

In previous studies [1–10], some research about parking lot detection, there was no mention of the influence of weather or conditions when taking digital images in such conditions as night or rain. In a way, this certainly affects digital images captured as data. In addition, the results of previous studies showed that the accuracy value was not optimal and can still be increased by observing the influence of the surrounding conditions or the weather.

Some researchers using support vector machines (SVM) to classify and detect empty parking lots [2, 5, 7, 8, 10]. From the observations that have been made, it is known that many parameters determine the SVM method to get the best accuracy performance. Therefore, the best parameter value search method is proposed to get the best accuracy performance. The best parameter values can achieve using a genetic algorithm (GA). GA is suitable for solving complex

Received: August 7, 2021. Revised: September 13, 2021.

problems using several genetic operations such as operator crossover and operator mutation. GA can help SVM get better parameter values, including penalty factors, kernel base radial function optimisation, and other parameters [11]. However, to obtain the characteristics of each land before SVM-based classification, generally, these methods utilise features based on colour histograms [8], areas and edges [10]. Because these features have weaknesses, especially when the land position undergoes translation or rotation, this study proposes an SVM classification method using GA based on gray level co-occurrence matrix (GLCM) features.

The contribution of this study was validated by comparing the generated model of empty parking lot detection using computer vision on public datasets such as the CNR parking dataset to (1) show the effect of different weather on detection performance, (2) analyse the performance of detection using a proposed GA to choose the parameter on SVM method, and (3) determine the model with the best performance result. The content is organised as follows. In section 2, related research on this research is highlighted. section 3 presents the proposed method, and section 4 discusses the result obtained from this research and explains the evaluation conducted on the model. Finally, the conclusion is presented in Section 5.

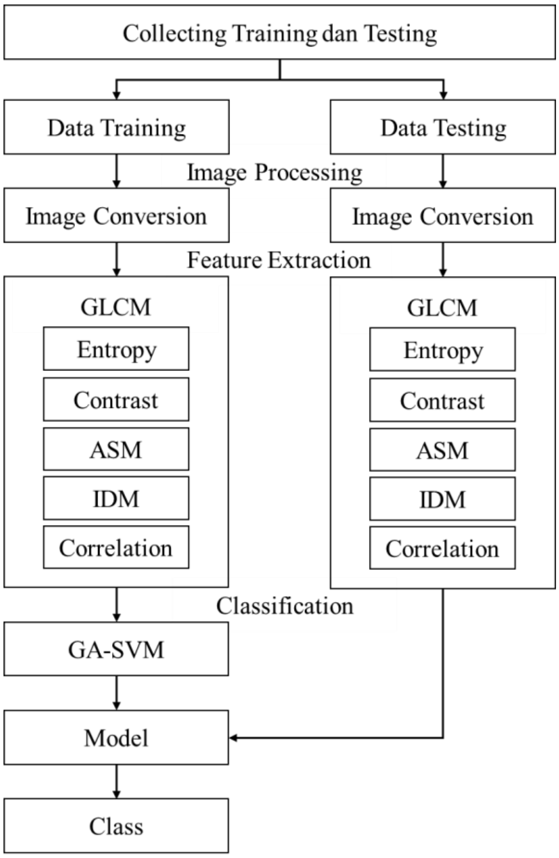

Figure. 1 Proposed method

2. Related research

Researchers have conducted many studies on empty parking lot detection to estimate whether four-wheeled vehicles or cars can occupy the parking space. Various methods are proposed in various research, from existing methods to those that offer new methods [1–10]. Dong [1] proposed a ranking algorithm to find parking efficiently. The algorithm is based on comprehensive public information from location prices, total space, and others. Nakazato and Namerikawa [9] proposed an intelligent system optimally allocating parking spaces using a matching base. Chen [3] proposed using the Dijkstra method to optimise the Floyd and ant colony algorithms. However, this study [3] did not show the results of the proposed method.

True [8] proposes a support vector machine (SVM) method based on the radial basis function (RBF) kernel to detect empty parking, and the results show an accuracy performance of 91 % with data that is influenced by morning and night. Wu [5] proposed the SVM based RBF and the Markov random field (MRF) methods to detect vacant parking lots. The results of the accuracy performance of the proposal show an accuracy of 93.52 %. Liu [7] proposed SVM and MRF methods similar to Wu [5]. Kabak and Turgut [10] proposed the SVM method with a linear kernel to detect empty parking from satellite images. The training results show that the performance is 94.29 %, and the testing results are 93.18 %. De Almeida [2] proposed an SVM classification method with texture-based feature extraction methods, and the training results reach a 99 % accuracy rate, and testing results reach an 89 % accuracy level. These results are obtained when applied to a data set with an illumination variance related to sunny, cloudy, and rainy days.

From these several studies, not many use datasets affected by sunny, cloudy, and rainy days. Only De Almeida [2] proposed using data affected by these environmental conditions. While the suggestions they gave still provide shortcomings in terms of parameter determination as in the SVM method, so the approach we chose was to offer the proper parameter determination by utilising GA.

3. Proposed method

This section shows the steps that the researcher took in this research. In the research method proposed by the researcher, the first step is to collect data. Then from the data, the conversion of RGB image data is converted into a grayscale image. Feature extraction

 DOI: 10.22266/ijies2021.1231.42

Received: August 7, 2021. Revised: September 13, 2021.

using gray level co-occurrence matrix (GLCM) is performed to obtain features in the form of the angular second moment (ASM), contrast, different inverse moment (IDM), entropy, and correlation [12]. GLCM feature is classified using the support vector machine (SVM) method. The Genetic Algorithm (GA) technique is used to find the best performance from the accuracy generated by the classification method. The best classification results produce a model, which is called SVMGA modelling. After the SVMGA modelling is obtained, the SVMGA model is tested, and the classification results are validated.

3.1 Data collection

The data used in this study is a public dataset obtained from Almeida [2]. The main problem involving the classification of parking spaces is the lack of consistent and reliable datasets. In the latest version of this dataset, images of parking spaces are taken from different parking lots with varying weather conditions, namely sunny, cloudy and rainy conditions. One of the advantages of using that dataset is that each parking space image of the parking area has been manually checked and segmented based on the situation (empty or occupied) and the weather conditions observed during image acquisition (sunny, cloudy, or rainy). An example of an image in the dataset is shown in Fig. 2.

The data provides in the form of images of parking lots in open spaces. About 10 % of the original data set will be used in this study, around 326 images consisting of 3 (three) available parking lots with sunny, cloudy, and rainy weather conditions with jpg extension. Each open parking lot image is segmented into as many as 224 parking space segments. Then it will convert the segmented image from RGB to a grayscale image. Total 326 images were divided into three weather conditions, with 101 images of sunny conditions, 110 cloudy conditions, and 115 rain conditions (Table 1). Then, it was separated into 2 with a ratio of 70 % for training data and 30 % for testing data randomly. In advance, the

Figure. 2 Image before segmentation

Table 1. Dataset description

No	Specification	Description
1.	Image Extension	.jpg
2.	Amount	326 parking space images
3.	Condition	sunny, cloudy, and raining
4.	Time	06.00 – 18.30 local time

Figure. 2 Parking area raw image

Figure. 3 Segmented empty and filled parking space

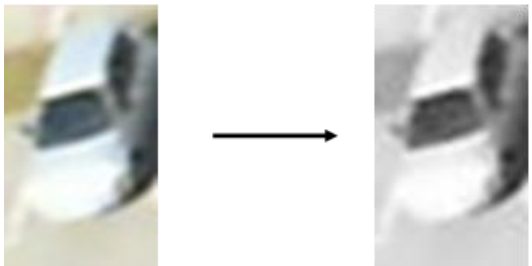

Figure. 4 Grayscale image for filled parking space

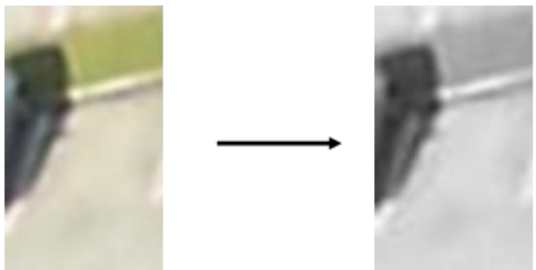

Figure. 5 Grayscale image for empty parking space

data should be pre-processed using segmentation and feature extraction techniques.

3.2 Segmentation

An XML file is included in the downloaded dataset for the segmentation process in the application to be run. This research is doing the segmentation process using the XM file that has been provided from the website. An example of segmentation using the attached XML file for a filled and empty parking lot is shown in Fig. 3 and Fig. 4.

3.3 Gray level co-occurrence matrix feature

Feature extraction uses grayscale images from RGB images, as shown in Fig. 5 and Fig. 6. The grayscale image is extracted using gray level co-occurrence matrix (GLCM) to obtain features such as the value of energy (ene), entropy (ent), contrast (con), homogeneity (hom), and correlation (cor). Each GLCM feature will be calculated based on angles of 0^0, 45^0, 90^0, and 135^0 with a distance pixel of 1. The angle is used to get the GLCM value, and feature calculations on each segmented parking lot are carried out. To extract the GLCM features in the filled parking slots according to Eq. (1) based on the GLCM formula.

$$GLCM_{\vec{r}}(i,j) = (x_1,y_1),(x_2,y_2) \in (N_x,N_y) \times (N_x,N_y)$$
$$f(x_1,y_1) = j^{\vec{r}} = \overrightarrow{(x_2 - x_1, y_2 - y_1)} \quad (1)$$

Table 2. Matrix of GLCM element

Value	Index (i, j)	Amount	Total Element
1	(41,49),(43,59),(44,43),...... ,(255,247)	980	980
2	(58,63),(66,58),(67,70),...... ,(255,254)	111	222
3	(166,166),(207,214),(210,211),.....,(255,251)	34	102
4	(207,207),(210,212),(211,211),.....,(252,253)	15	60
5	(211,210),(211,212),(215,214),.....,(252,252)	7	35
6	(212,213),(214,213),(214,215),.....,(255,253)	6	36
7	(219,218)	1	7
8	(212,212),(213,214),(217,217),.....,(254,255)	5	40
9	(218,217),(253,253)	2	18
10	(210,210),(254,254)	2	20
11	(213,213)	1	11
13	(219,219)	1	13
14	(217,218)	1	14
15	(218,219)	1	15
17	(255,255)	1	17
Total Element			**1590**

The first calculation determines the x and y values, where x is for width and y is for height. The sample image size that will be calculated has a width of 32 and a height of 55. The image matrix shows that the highest value of the matrix elements is 255, so the GLCM matrix to be formed is 255×255, and the GLCM matrix is calculated based on Eq. (2) until Eq. (6) [12], respectively. The total element of matrix GLCM can be seen in Table 2. After getting the feature value, normalisation is doing by looking at the value of the GLCM features. The distribution of distance or weight between the five features is very far—normalisation changes the GLCM feature values to a value between 0 and 1.

$$ene = \sum_{k=0}^{L-1}\sum_{l=0}^{L-1}\{P(k,l)\}^2 \quad (2)$$

$$ent = -\sum_{k=0}^{G-1}\sum_{l=0}^{G-1} P(k,l) \times Log\big(P(k,l)\big) \quad (3)$$

$$con = \sum_{n=0}^{G-1} n^2\{\sum_{k=0}^{G}\sum_{l=0}^{G} P(k,l)\}\{n = |k-l| \quad (4)$$

$$hom = \sum_{k=0}^{G-1}\sum_{l=0}^{G-1} \frac{1}{1+(k-l)^2} P(k,l) \quad (5)$$

$$cor = \frac{\sum_{k=0}^{G-1}\sum_{l=0}^{G-1}(k,l)(P(k,l) - \mu_k{}'\mu_l{}')}{\sigma_k{}'\sigma_l{}'} \quad (6)$$

The correlation is obtained using Eq. (7).

$$\begin{aligned} P_x(k) &= \sum_{l=0}^{G-1} P(k,l) \\ P_y(l) &= \sum_{k=0}^{G-1} P(k,l) \\ \mu'_k &= \sum_{k=0}^{G-1}\sum_{l=0}^{G-1} k \times P(k,l) \\ \mu'_l &= \sum_{k=0}^{G-1}\sum_{l=0}^{G-1} l \times P(k,l) \\ \sigma'_k &= \sum_{k=0}^{G-1}\sum_{l=0}^{G-1} P(k,l)(k - \mu'_l)^2 \\ \sigma'_l &= \sum_{k=0}^{G-1}\sum_{l=0}^{G-1} P(k,l)(l - \mu'_l)^2 \end{aligned} \quad (7)$$

3.4 Support vector machine

Support vector machine (SVM) is a learning system that uses a hypothetical rating in linear functions in a high-dimensional feature space trained with a learning algorithm based on optimisation theory by implementing a learning bias derived from statistical theory [13]. SVM classification attempts to find the best hyperplane that functions as a separator of two data classes in the input space. It can find the best hyperplane between the two classes by measuring the hyperplane margin and finding the maximum point. Margin is the distance between the hyperplane and the closest data from each category, while the data were closest to the hyperplane or support vector.

Non-linear SVM maps the training sample from the input space to the higher-dimensional feature space via the mapping function 0 in the kernel

Received: August 7, 2021. Revised: September 13, 2021.

function [14]. SVM finds the optimal separating hyperplane with a maximum margin by solving the optimisation problem, which is done by finding the saddle point of the Lagrange function. To find the optimal hyperplane, a dual Lagrangian $L_D(\alpha)$ must be maximised against non-negative (α). The equation of dual Lagrangian can be seen in Eq. (8). The inner product can be replaced with the kernel function in Eq. (9). Non-linear SVM dual Lagrangian in Eq. (8) is similar to linear generalised.

$$\left(\Phi(\mathrm{x_i}) \cdot \Phi(\mathrm{x_i})\right) \coloneqq k(x_i, x_j) \tag{8}$$

$$L_D(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \tag{9}$$

Where $\alpha_i \geq 0$ and $i = 1, 2, \dots\dots, m$, also the $\sum_{i=1}^{m} \alpha_i y_i = 0$.

The optimisation model is solved using the optimisation method inseparable cases. The hyperplane has the form Eq. (9) and depends on the kernel, and the b bias can be implicitly part of the kernel function. The bias is accommodated in the kernel function. The non-linear support vector classifier can be shown as Eq. (10).

$$f(x, a', b') = \sum_{i=1}^{m} y_i\, \alpha_i' \left(\Phi(x_i) \cdot \Phi(x_i)\right) + b' \quad \text{become} \quad f(x, a', b') = \sum_{i=1}^{m} y_i\, \alpha_i' k(x_i, x) + b' \tag{10}$$

$$f(x, a', b') = \sum_{i \in sv}^{m} y_i\, \alpha_i' \left(\Phi(x_i) \cdot \Phi(x_i)\right) \quad \text{become} \quad f(x, a', b') = \sum_{i \in sv}^{m} y_i\, \alpha_i' k(x_i, x) \tag{11}$$

Kernel functions can be used as polynomial based on Eq. (12), radial basis function (RBF) based on Eq. (13), and sigmoid kernel based on Eq. (14). Kernel parameters in the kernel functions should be set appropriately in order to improve classification accuracy.

$$k(x_i, x_j) = \left(1 + x_i \cdot x_j\right)^d \tag{12}$$

$$k(x_i, x_j) = \exp\left(-\gamma \left\| x_i - x_j \right\|^2\right) \tag{13}$$

$$k(x_i, x_j) = \tanh(k x_i \cdot x_j - \delta) \tag{14}$$

3.5 Optimization using genetic algorithm

A genetic algorithm (GA) is an adaptive optimisation search method based on genetics in biological systems. GA works with a set of candidate solutions called a population. GA obtains the optimal solution after a series of iterative calculations. GA generates successive populations of solutions represented by chromosomes. Search characteristics in exploitation and exploration can efficiently handle search space and provide less opportunity to find local optimal solutions [15].

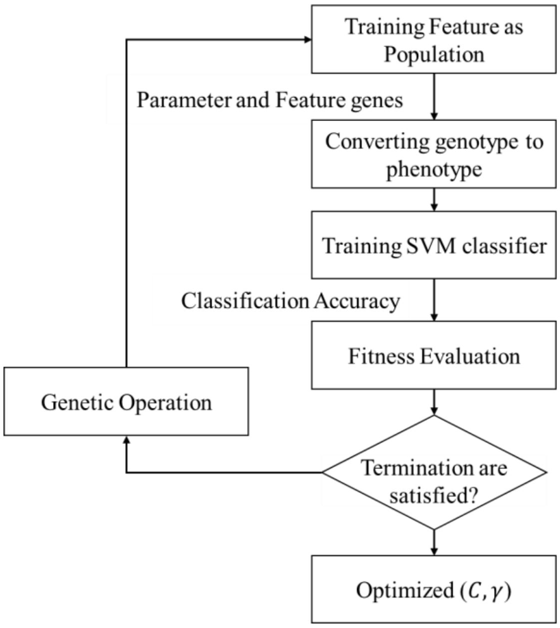

Figure. 6 Architecture for SVMGA

The fitness function evaluates the quality of the solution in the iteration [16]. Crossover and mutation functions randomly affect the fitness value. Chromosomes are selected for reproduction by evaluating their fitness values. The best chromosomes will likely be selected using the roulette wheel or selection method. In this study, GA was used to obtain the best accuracy results from the SVM classification method, with the details shown in Fig. 7.

3.6 Performance evaluation

Evaluation of the performance of this study was carried out by calculating the value of accuracy. This value is the result of the correct classification of all data obtained. Values are obtained using Eq. (15), with t as the correct sample data and n as the total sample data.

$$accuracy = \frac{t}{n} \times 100 \tag{15}$$

4. Research design

This research was conducted using datasets [2], and applied approach using texture-based feature extraction also SVM classification method based [5], [8]. The experiments were conducted to get the best accuracy in detecting empty parking. A comparison is made with previous studies to observe the best accuracy. The methods are compared by comparing proposals [5], [8] compared to this approach.

The parameters used are based on SVM parameters such as C and γ parameters on RBF. The SVM grid search method is used to find the optimal parameter. In this method, the value range limitation for C values ranges from 2^{-5} to $2^{10,}$ and γ ranges from 2^{-9} to 2^{15}. The range limitation is carried out due to the consideration of program execution time. The value of the range parameter C is divided into two scales, and the value γ is divided into four scales. C and γ parameters in SVM are performed on to the data based on weather conditions, like sunny, cloudy, and rainy.

This research using a genetic algorithm (GA) to optimise the optimal value for the hyperplane on the SVM parameter. The chromosome values GA using the lower and upper limits from the optimal parameter value range for the SVM parameter. The fitness value is an objective function of the genetic algorithm that determine by accuracy. The GA step, such as initialisation, selection, crossover, mutation, and elitism, are used in the experiment. The parameter GA used crossover probability as 0.4 and mutation probability as 0.1. All experiment in this study is done using MATLAB.

5. Result and discussion

5.1 Classification using SVM

Using 10-fold cross-validation, the accuracy results of these experiments based on the weather are shown in Table 3. The figure shows that C optimal average accuracy lies in the range 2^5 - 2^{10} and for γ is in the value range 2^{-3} - 2^3. The average accuracy obtained using parameter C in the value range of 2^5 - 2^{10} and parameter γ in the value range of 2^{-3} - 2^3 is 92.04 % for sunny weather, 96.83 % for cloudy weather, and rainy weather conditions. 95.31 %. Sunny weather conditions are the minimum parameter that needs to improve the accuracy. The optimal value as a reference in optimising the SVM parameters, 12 experiments were carried out using the range of values obtained.

Table 3. Classification result using grid search

Parameter Range		Average Accuracy (%)		
C	γ	Sunny	Cloudy	Raining
$2^{-5} – 2^0$	$2^{-9} – 2^{-3}$	71,05	72,70	75,17
	$2^{-3} – 2^3$	88,29	92,32	90,63
	$2^3 – 2^9$	88,57	91,79	90,69
	$2^9 – 2^{15}$	66,64	64,49	69,51
$2^0 – 2^5$	$2^{-9} – 2^{-3}$	85,69	88,77	88,53
	$2^{-3} – 2^3$	90,51	95,25	93,72
	$2^3 – 2^9$	92,14	96,69	95,36
	$2^9 – 2^{15}$	79,59	78,79	81,23

Table 4. The experiment of SVM on training data

Experiment no.	Optimised Parameter		Accuracy (%)
	C	γ	
1	2^0	2^{-3}	91,06
2		2^{-1}	92,67
3		2^1	93,50
4		2^3	93,50
5	2^2	2^{-3}	92,04
6		2^{-1}	93,39
7		2^1	93,39
8		2^3	93,50
9	2^4	2^{-3}	92,77
10		2^{-1}	93,50
11		2^1	93,60
12		2^3	93,55

From the experiments carried out on the training data as shown in Table 4, the optimal value occurs at C = 2^4 and γ = $2^{1,}$ which occurs in the 11 th experiment resulting in an accuracy of 93.60 %.

5.2 Parameter optimization using SVMGA

The optimisation is needed to get a more optimal output value in the testing process that will be carried out later. From the model obtained, the prediction becomes more optimal in the test data set. The optimisation is carried out on the SVM parameters, namely C and γ, while the fitness value is obtained from the accuracy value. The optimal hyperplane can determine the fitness value for SVM with the RBF kernel function (based on Eq. (13)). The GA stages consisting of initialisation, selection, crossover, mutation, and elitism are shown as follows.

5.2.1. Initialization step

In this process of initialisation, the random value is generated with a predetermined range of values. The values were $0 \leq \gamma \leq 2.5$ and $8 \leq C \leq 16$, and it was used to limit the value of genes on chromosomes (are as shown in Table 5).

The chromosome bit length is determined then initialises as many as N chromosomes containing the SVM parameter value. The chromosomes formed will undergo all the processes in the genetic algorithm, including selection, crossing over, mutation, and elitism to obtain parameters that will produce optimal accuracy values. Suppose that ten

Table 5. Chromosome initialisation

Chromosomes	C	γ
	22	2

Table 6. Fitness value for each chromosome

Chromosome	Gen		Fitness
	C	γ	
1	11,605	1,232	94,29
2	19,584	1,639	94,55
3	20,139	1,116	94,76
4	5,891	2,285	94,18
Chromosome	Gen		Fitness
	C	γ	
5	27,740	1,196	94,13
6	9,498	1,611	94,08
7	13,557	2,458	94,03
8	25,682	1,779	94,18
9	30,861	2,886	93,82
10	4,567	2,147	93,82
Total fitness			941,84

chromosomes are determined at an early stage, illustrated as shown in Table 6.

5.2.2. Selection step

The roulette wheel selection method is the method chosen in this study to select populations that have been formed at the initialisation stage. The roulette wheel is one of the methods in the selection stage to determine the chromosomes of parents or parents who survive for the next generation. By looking at the fitness value, the chromosomes that are formed will be selected for the next generation. The probability of the *I* chromosome being selected at the fitness value f_i^n (as shown in Table 7).

The roulette wheel selection method selects the chromosomes with the chance that they are proportional to their cumulative probability value. The cumulative probability value of a chromosome is greater than the chromosome that will be selected to survive (as in Table 8). After that, generate a random number $0 \leq R \leq 1$ as much as the population (as shown in Table 9).

If $R_k < C_k$ then the k chromosome will be selected as a candidate parent with conditions $C_{k-1} < R_k < C_k$. As of the prospective parents with selected chromosomes as Table 10.

5.2.3. Crossover step

In the crossover stage, two-parent chromosomes are carried out to produce offspring or children. Parent chromosomes are randomly selected by determining the crossover probability value. In this study, the crossover probability as big as 0.4, so four chromosomes of the candidate parent are used as the parent. The method of crossing over or Crossover in this study is real numbers using the whole arithmetic crossover calculation by determining α as big as

Table 7. Probability of each chromosome

Chromosome	Gen		Fitness	Probability
	C	γ		
1	11,605	1,232	94,29	0,1001
2	19,584	1,639	94,55	0,1004
3	20,139	1,116	94,76	0,1006
4	5,891	2,285	94,18	0,1000
5	27,740	1,196	94,13	0,0999
6	9,498	1,611	94,08	0,0999
7	13,557	2,458	94,03	0,0998
8	25,682	1,779	94,18	0,1000
9	30,861	2,886	93,82	0,0996
10	4,567	2,147	93,82	0,0996
Total fitness			941,84	

Table 8. Probability cumulative of each chromosome

Chromosome	Fitness	Probability	Probability Cumulative
1	94,29	0,1001	0,1001
2	94,55	0,1004	0,2005
3	94,76	0,1006	0,3011
4	94,18	0,1000	0,4011
5	94,13	0,0999	0,5011
6	94,08	0,0999	0,6009
7	94,03	0,0998	0,7008
8	94,18	0,1000	0,8008
9	93,82	0,0996	0,9004
10	93,82	0,0996	1,0000

Table 9. Generated random value of each chromosome

Chro mo some	Fitness	Probability	Probability Cumulative	Random Value
1	94,29	0,1001	0,1001	0,8594
2	94,55	0,1004	0,2005	0,9631
3	94,76	0,1006	0,3011	0,1478
4	94,18	0,1000	0,4011	0,5759
5	94,13	0,0999	0,5011	0,0475
6	94,08	0,0999	0,6009	0,6347
7	94,03	0,0998	0,7008	0,3805
8	94,18	0,1000	0,8008	0,4510
9	93,82	0,0996	0,9004	0,2318
10	93,82	0,0996	1,0000	0,9735

Table 10. Chosen chromosome as parent candidate

Parent	Random Value	Chromosome
1	0,0475	5
2	0,1478	3
3	0,2318	9
4	0,3805	7
5	0,4510	8
6	0,5759	4
7	0,6347	6
8	0,8594	1
9	0,9631	2
10	0,9735	10

0.6176. The chromosomes selected as a parent are chromosomes 5 and 7, 3 and 8 (in Table 11).

Then, the crossover process was carried out using Eq. (15) for Parent a and Eq. (16) for Parent b. The total population has been seen in Table 12.

$$C_1 = \alpha.\bar{x}(1-\alpha).\bar{y} \tag{16}$$

$$C_2 = \alpha.\bar{y}(1-\alpha).\bar{x} \tag{17}$$

Crossover result on first pair chromosome (5, 7)

$C1$	0,6176×27,740+(1-0,6176) ×13,557 = **22,316**	0,6176×1,196+(1-0,6176) ×2,458 = **1,6785**
C_2	0,6176×13,557+(1-0,6176) ×27,740 = **24,782**	0,6176×2,458+(1-0,6176) ×1,196 = **1,9754**

Crossover result on second pair chromosome from number (3, 8)

C_3	0,6176×20,139+(1-0,6176)×25,682 = **22,258**	0,6176×1,116+(1-0,6176)×1,779 = **1,3695**
C_4	0,6176×25,682+(1-0,6176)×20,139 = **23,562**	0,6176×1,779+(1-0,6176)×1,116 = **1,5254**

Table 11. Selected pair of chromosome

Chromosome	C	γ
5	27,740	1,196
7	13,557	2,458
Chromosome	C	γ
3	20,139	1,116
8	25,682	1,779

Table 12. Crossover result on population

Chromosome	Gen		Fitness
	C	γ	
P_1	11,605	1,232	94,29
P_2	19,584	1,639	94,55
P_3	20,139	1,116	94,76
P_4	5,891	2,285	94,18
P_5	27,740	1,196	94,13
P_6	9,498	1,611	94,08
P_7	13,557	2,458	94,03
P_8	25,682	1,779	94,18
P_9	30,861	2,886	93,82
P_{10}	4,567	2,147	93,82
C_1	22,316	1,678	94,29
C_2	24,782	1,975	94,13
C_3	22,258	1,369	94,83
C_4	23,562	1,525	94,50

5.2.4. Mutation step

In the mutation process, probability value (P_m) need to determine before generating random numbers with a value between 0 and 1 for each gene on the chromosome. For genes whose random number is below the value of P_m, the gene will be mutated within a predetermined range of values. Suppose that the value of P_m as 0.1 while the range of values is 0 ≤ γ ≤ 2.5 and 8 ≤ C ≤ 16. The picture above shows that gene C in C3 and gene γ in C4 have mutations because the random numbers are smaller than P_m, namely 0.008 < 0.1 and 0.089 < 0.1. The value on the gene is replaced by the random number value obtained. The illustration of mutation has been shown in Fig. 8.

5.2.5. Elitism step

Elitism serves to maintain chromosomes which has the highest fitness value derived from the calculation of the parameters. In this study, the parameters are C and γ aims to maintain the optimal

Chromosome	Gen	
	C	γ
Random Value	0,008	0,145
C_3	22,258	1,369
Random Value	0,235	0,089
C_4	23,562	1,525

C_3 and C_4 become C_3' and C_4'

C_3'	10,326	1,369
C_4'	23,562	1,423

Figure 7. Mutation process

Table 13. Overall population based on fitness

Chromosome	Gen		Fitness
	C	γ	
$C_{3'}$	10,326	1,369	94,77
P_3	20,139	1,116	94,76
P_2	19,584	1,639	94,55
$C_{4'}$	23,562	1,423	94,55
P_1	11,605	1,232	94,29
C_1	22,316	1,678	94,29
P_4	5,891	2,285	94,18
P_8	25,682	1,779	94,18
P_5	27,74	1,196	94,13
C_2	24,782	1,975	94,13
P_6	9,498	1,611	94,08

Chromosome	Gen		Fitness
	C	γ	
P_7	13,557	2,458	94,03
P_9	30,861	2,886	93,82
P_{10}	4,567	2,147	93,82

value in each population formed. In this study, the population maintained was 70 % of the total initial population (14 chromosomes), as shown in Table 13.

The second generation has the following initial population as in Table 14. The process is carried out repeatedly, starting from the selection stage to Elitism until the stopping criteria that have been determined are met.

5.3 Evaluation using SVMGA

Evaluation is done by preparing all datasets that have sunny, cloudy, rainy conditions into one. The best parameter results are used in the test. For each test data, there are 224 segmented parking lot images. An example of an image before classification is shown in Fig. 9 and Fig. 10. Fig. 10 shows that the 224 parking are classified 157 parking spaces can be classified as content, and 67 parking spaces are classified as empty. From the 157 parking lots, 11 parking spaces are misidentified, and out of the 67 parking lots classified as empty, there is no parking that was misidentified, and the accuracy is 94,64%.

Tests are also carried out individually for each condition (are shown in Table 15). The data applied experienced sunny, cloudy, and rainy weather conditions. In the experiment, testing this proposal gave an average performance of 96.99 %, which was an increase of 6.99 % compared to the study of True [8]. Compared with Wu [5], the approach

Table 14. 2nd Generation population

Chromosome	Gen		Fitness
	C	γ	
P_1	10,326	1,369	94,77
P_2	20,139	1,116	94,76
P_3	19,584	1,639	94,55
P_4	23,562	1,423	94,55
P_5	11,605	1,232	94,29
P_6	22,316	1,678	94,29
P_7	5,891	2,285	94,18
P_8	25,682	1,779	94,18
P_9	27,74	1,196	94,13
P_{10}	24,782	1,975	94,13

Figure. 8 Image example before classification

Table 15. Average classification accuracy based on weather condition

Weather	Number of Data	Accuracy
Sunny	25	96,78%
Cloudy	25	98,03%
Rainy	15	96,16%
Test Data Total	65	
Mean Accuracy		96,99%

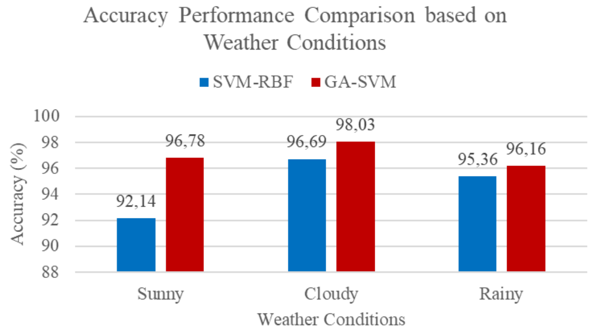

Figure. 9 Result comparison between SVM-RBF [5], [8] and SVMGA (proposed method)

Figure. 10 Misclassification example on sunny weather condition

performance gives an increase of 3.47 %. Meanwhile, when compared with the approach of Kabab [10], this study provides an increase of 6.81 %. According to the dataset, the previous research was applied based on the weather conditions (Fig. 11). The highest accuracy occurred when applied to cloudy weather conditions. The accuracy reached 98.03 % is higher 1.34 % than previous methods. In sunny weather, the accuracy is 96.78 % is higher at 4.64 % than previous methods. Meanwhile, in rainy weather conditions, 96.16 % is higher than 0.80 % than the previous method.

The highest average given in this study is in cloudy weather conditions (Fig. 11), which is 98.03 % and still misclassified. Misclassification is due to the object or position of the parking vehicle crossing the parking area dividing line. In sunny weather conditions, the light intensity is too high and affecting the classification results. In contrast, rainy weather occurs when the parking slot is submerged.

The approach algorithm sometimes cannot correctly distinguish whether the vehicle is parked or the parking area is wide where the lane position exists in the middle of the parking lot. It is different when the weather is cloudy with average light intensity, so during the classification process, it can run optimally. Misclassification occurs in the blue box, as shown in Fig. 12.

That figure shows that the classification error is due to the difference in light intensity, which light detected in part of the parking area. It looks like there is an intersection in the image. In Fig. 13, an object looks like a human in the parking area, and the algorithm detects it as a vehicle parked in the parking area in cloudy weather. In Fig. 14, the classification error is caused by a pool of water that looks like an object in the parking lot for rainy weather.

5.4 Evaluate with another dataset

The reliability of the SVMGA algorithm model that has been formed is tested on other parking lots. The public dataset comes from the CNR parking dataset, can be downloaded from the CNR webpage http://www.cnrpark.it/. The data contains 38 images divided into two weather categories, sunny and overcast, with 19 images respectively.

Table 16 shows the results of the average classification test using the SVMGA model on the CNR parking lot test data for sunny and overcast weather conditions. The SVMGA algorithm works very well on the CNR parking lots dataset. The optimal accuracy value occurs in sunny weather with an average accuracy of 94.84 % and overcast weather with an average accuracy value of 93.88 %. In general, misclassification is caused by objects and puddles in the parking lot, as shown in Fig. 15 and Fig. 16.

Figure. 11 Misclassification example on cloudy weather condition

Figure. 12 Misclassification example on rainy weather condition

Table 16. Test result on CNR dataset

Weather	Sunny	Overcast	Total Data	Mean Accuracy
Number of Data	19	19	38	
Accuracy	94,84%	93,88%		94,36%

Figure. 13 Misclassification on CNR dataset (sunny)

Figure. 14 Misclassification on CNR dataset (overcast)

6. Conclusion

In this study, the detection of empty parking lots by previous researchers gave a varied performance. The SVMGA approach provides an accuracy performance of 96.99 %, which is 2.65 % better than previous studies. The experimental results also show that this proposed method is successful when applied to other public datasets with an accuracy performance amount of 94.36 %. It is hoped that this research can be tested further and applied directly to benefit many people in the future.

Conflicts of interest

Following the International Journal of Intelligent Engineering and Systems, policy, and my ethical obligation as a researcher, I am reporting that this paper has not been published previously, has not been copyrighted, has not been submitted elsewhere. I have disclosed those entirely to the International Journal of Intelligent Engineering and Systems and get approval from all authors to manage any potential conflicts from that research.

Author Contributions

Conceptualisation, Yuslena Sari and Hargokendar Suhud; methodology, Yuslena Sari and Hargokendar Suhud; software, Hargokendar Suhud; validation, Andreyan Rizky Baskara, Ricardus Anggi Pramunendar, and Iphan F. Radam; formal analysis, Yuslena Sari, Hargokendar Suhud and Ricardus Anggi Pramunendar; investigation, Yuslena Sari and Hargokendar Suhud; resources, Yuslena Sari and Hargokendar Suhud; data curation, Hargokendar Suhud; writing—original draft preparation, Yuslena Sari; writing—review and editing, Andreyan Rizky Baskara, Ricardus Anggi Pramunendar, and Iphan F. Radam; visualisation, Hargokendar Suhud; supervision, Andreyan Rizky Baskara, Ricardus Anggi Pramunendar, and Iphan F. Radam; project administration, Yuslena Sari; funding acquisition, Yuslena Sari.

Acknowledgements

The authors would like to thank the Universitas Lambung Mangkurat, STIKES Karya Husada, and Universitas Dian Nuswantoro for logistical and financial support.

References

[1] S. Dong, M. Chen, L. Peng, and H. Li, "Parking rank: A novel method of parking lots sorting and recommendation based on public information", In: *Proc. of 2018 IEEE International Conference on Industrial Technology*, pp. 1381–1386, 2018.

[2] P. R. L. D. Almeida, L. S. Oliveira, A. S. Britto, E. J. Silva, and A. L. Koerich, "PKLot – A robust dataset for parking lot classification", *Expert Systems with Applications*, Vol. 42, No. 11, pp. 4937–4949, 2015.

[3] R. Chen, X. Hu, and W. Mu, "Research on parking lot management system based on parking space navigation technology", In: *Proc. of 2020 IEEE International Conference on Power, Intelligent Computing and Systems,* pp. 773–777, 2020.

[4] Y. C. Yeh and M. S. Tsai, "Impact simulation of PEV parking lots to power distribution systems", In: *Proc. of IEEE International Symposium on Industrial Electronics*, pp. 117–122, 2017.

[5] Q. Wu, C. Huang, S. Wang, W. Chiu, and T. Chen, "Robust Parking Space Detection Considering Inter-Space Correlation", In: *Proc. of 2007 IEEE International Conference on Multimedia and Expo*, pp. 659–662, 2007.

[6] O. T. T. Kim, N. H. Tran, C. Pham, T. Leanh, M. T. Thai, and C. S. Hong, "Parking Assignment: Minimizing Parking Expenses and Balancing Parking Demand among Multiple Parking Lots", *IEEE Transactions on Automation Science and Engineering*, Vol. 17, No. 3, pp. 1320–1331, 2020.

[7] Y. H. Liu, K. L. Ong, V. C. S. Lee, and Y. P. P. Chen, "Processing multiple image streams for real-time monitoring of parking lots", *International Journal on Computer, Consumer and Control*, Vol. 1, No. 1, pp. 23–32, 2011.

[8] N. True, "Vacant parking space detection in static images", *University of California, San Diego*, 2007, [Online]. Available: http://cseweb.ucsd.edu/classes/wi07/cse190-a/reports/ntrue.pdf.

[9] T. Nakazato and T. Namerikawa, "Parking Lot Allocation Based on Matching Theory using Prediction-based Optimal Vehicle Routing", In: *Proc. of 19th International Conference on Control, Automation and Systems*, pp. 1004–1009, 2019.

[10] M. O. Kabak and O. Turgut, "Parking Spot Detection from Aerial Images", *Seven*, No. 5229284, pp. 1–3, 2010.

[11] C. Wei and H. M. Yuan, "An improved GA-SVM algorithm", In: *Proc. of the 2014 9th IEEE Conference on Industrial Electronics and Applications*, pp. 2137–2141, 2014.

[12] R. A. Pramunendar, D. P. Prabowo, D. Pergiwati, Y. Sari, P. N. Andono, and M. A. Soeleman, "New Workflow for Marine Fish Classification Based on Combination Features and CLAHE Enhancement Technique", *International Journal of Intelligent Engineering and Systems*, Vol. 13, No. 4, pp. 293–304, 2020.

Received: August 7, 2021. Revised: September 13, 2021.

[13] D. A. Gustian, N. L. Rohmah, G. F. Shidik, A. Z. Fanani, R. A. Pramunendar, and Pujiono, “Classification of Troso Fabric Using SVM-RBF Multi-class Method with GLCM and PCA Feature Extraction”, In: *Proc. 2019 International Seminar on Application for Technology of Information and Communication: Industry 4.0: Retrospect, Prospect, and Challenges, iSemantic 2019*, pp. 7–11, 2019.

[14] C. L. Huang and C. J. Wang, “A GA-based feature selection and parameters optimizationfor support vector machines”, *Expert Systems with Applications*, Vol. 31, No. 2, pp. 231–240, 2006.

[15] M. S. Gibbs, H. R. Maier, G. C. Dandy, and J. B. Nixon, “Minimum number of generations required for convergence of genetic algorithms”, In: *Proc. of 2006 IEEE Congress on Evolutionary Computation, CEC 2006*, pp. 565–572, 2006.

[16] E. Noersasongko, F. T. Julfia, A. Syukur, P. R. A. Pramunendar, and C. Supriyanto, “A Tourism Arrival Forecasting using Genetic Algorithm based Neural Network”, *Indian Journal of Science and Technology*, Vol. 9, No. 4, pp. 3–7, 2016.