2019 6th International Conference on Electric Vehicular Technology (ICEVT)
November 18-21, 2019, Bali, Indonesia

Road Crack Detection using Support Vector Machine (SVM) and OTSU Algorithm

Yuslena Sari
Department of Information Technology, Faculty of Engineering Universitas Lambung Mangkurat
Banjarmasin, Indonesia
National Center for Sustainable Transportation Technology
Indonesia
yuzlena@ulm.ac.id

Puguh Budi Prakoso
Department of Civil Engineering, Faculty of Engineering Universitas Lambung Mangkurat
Banjarmasin, Indonesia
National Center for Sustainable Transportation Technology
Indonesia
puguh.prakoso@ulm.ac.id

Andreyan Rezky Baskara
Faculty of Engineering Universitas Lambung Mangkurat
Banjarmasin, Indonesia
National Center for Sustainable Transportation Technology
Indonesia
andreyan.baskara@ulm.ac.id

***Abstract*—Cracks are one type of pavement surface damages, whose assessment is very important for developing road network maintenance strategies, which aims to ensure the functioning of the road and driving safety. Existing methods for automatic crack detection depend mostly on expensive equipment and high maintenance and cannot divide the crack segments accurately. This paper discusses an automation method of classification and segmentation of asphalt pavement cracks. The goal of the research is to classify asphalt pavement cracks using the classification method of the Support Vector Machine (SVM) algorithm and segmentation method of the OTSU algorithm. The OTSU algorithm for segmentation has advantages in choosing the optimal threshold that is stable. This algorithm is proven to be more effective and stronger than conventional segmentation algorithms. For detection results, the proposed method achieves overall accuracy.**

***Keywords*—road crack, road crack detection, Support Vector Machine (SVM), OTSU**

I. Introduction

Asphalt pavement is an important part of national infrastructure. Therefore, an effective and efficient assessment of pavement conditions is an important concern for transport authorities in many countries to set maintenance schedules, approaches and budgets[1]–[3]. In the US, maintenance and rehabilitation of asphalt pavements require more than 17 billion dollars per year. The traditional sidewalk cracking detection system by the human eye is very expensive, requires a lot of energy, time, and subjective[4]. Because of the high demands for an intelligent pavement management strategy, the development of automatic pavement detection systems has received a lot of attention in the last decade.

A vehicle for monitoring road conditions is specially designed and equipped with positioning systems, cameras, laser scanners, pavement profiles, and accelerometers to collect pavement data[2][5]. However, for example in the North America, Tsai and Li's research shows that only 8 highway authorities utilize automatic crack detection in practice[6]. With the proliferation of technology, more transport authorities are using automated techniques to collect data. However, manual data processing is still dominating. In a recent study, Radopoulou and Brilakis[5] estimated that only 0.4% of inspections are automatic and the remaining 99.6% are manual.

Two-dimensional (2D) pavement images are the main data sources used in practice for the detection and segmentation of cracks. Automatic crack detection and segmentation based on 2D images is challenging because (1) low contrast between cracks and surrounding pavement, (2) complicated crack patterns, and (3) genomic intensity along cracks[6]–[8]. Based on the problems mentioned above, this paper presents a method for automatically detecting and classifying asphalt cracks. The aim of the study is to classify asphalt pavement cracks using the classification method of the Support Vector Machine (SVM) algorithm and segmentation method of the OTSU algorithm.

II. Related Work

The Intensity-thresholding method has been widely implemented for crack detection[9],[10]. Nevertheless, background lighting and pavement texture significantly affect the performance of this method, which results unreliable crack segmentation. In the case of images with a low noise signal ratio, the OTSU algorithm is widely applied by researchers today[11]. The application of a simple and efficient segmentation algorithm is actually able to produce satisfactory performance in detection[11]–[13].

Prasanna et al. [14] designed a histogram-based classification algorithm and applied it together with Support Vector Machines (SVM) to detect cracks on the concrete deck surface; the results on bridge data highlight the need to improve the accuracy of practical predictions. Nhat□Duc Hoang1 and Quoc□Lam Nguyen[15] compared classification algorithms using machine learning, the results showed that SVM had reached the highest level of classification accuracy (87.50%), followed by ANN (84.25%), and RF (70%). Gavilan et al[16] made a road crack detection system using Support Vector Machine (SVM), the result is a linear SVM based classifier can distinguish between 10 types of pavements that appear on Spanish roads. The SVM-based method that takes into account neighbouring pixel information was recently introduced by Ai et al[17]. Based on previous research works, this study proposes a pavement crack classification model using the SVM method with the OTSU algorithm as the segmentation method.

978-1-7281-2917-4/19/$31.00 ©2019 IEEE

III. METHODOLOGY

The following diagram in Fig.1 is a general description of the proposed method, consisting of three steps for crack detection. First the images that have been collected one by one will be segmented using the OTSU algorithm. Image segmentation is a process, which is aimed to obtain objects contained in the image or to divide the image into several regions, in which each object or region has a similar attribute. In images containing only one object, the object is distinguished from its background. After that, feature extraction is performed to get the characteristics of the image. The last stage is the classification by using the SVM method.

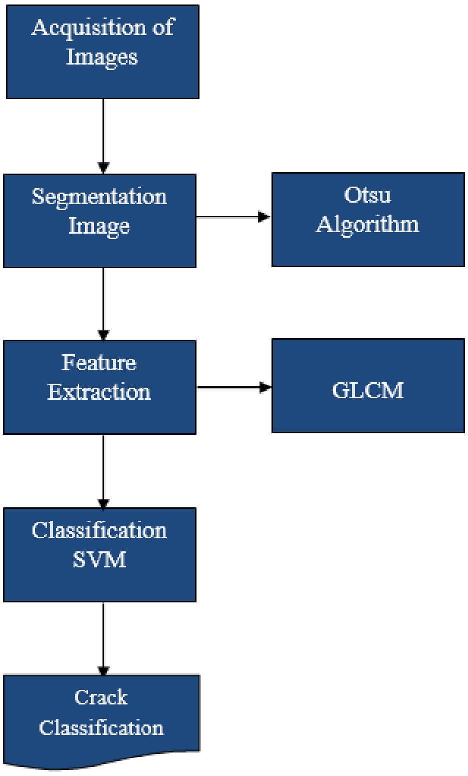

Fig. 1. Proposed Algorithm

A. *Data Collection*

The collected dataset was road image data in Banjarmasin, Indonesia. Image data is taken using a low-cost smart phone camera.

B. *Preliminary Data Processing*

The collected dataset is pre-processed by resizing the image to 256x100 pixels.

C. *Methods*

1) OTSU:

OTSU performs discriminant analysis by determining a variable by distinguishing between two or more groups naturally. The OTSU method starts with normalizing the histogram image as a discrete density probability function as follow:

$$p_r(r_q) = \frac{n_q}{n}, where\ q = 0, 1, 2, \ldots, L-1 \quad (1)$$

where n is the total number of pixels in the image, n_q is the number of pixels r_q, and L is the total number of image intensity levels.

In determining the value of T by maximizing between class variance is defined as follows:

$$\sigma_B^2 = \omega_o(\mu_0 - \mu_T)^2 + \omega_1(\mu_1 - \mu_T)^2 \quad (2)$$

where obtained from:

$$\omega_o = \sum_{q=0}^{k-1} p_q(r_q) \quad where\ \omega_1 = \sum_{q=k}^{L-1} p_q(r_q)$$

$$\mu_0 = \sum_{q=0}^{k-1} \frac{q p_q(r_q)}{\omega_o} \quad where\ \mu_1 = \sum_{q=k}^{L-1} \frac{q p_q(r_q)}{\omega_o}$$

$$\mu_T = \sum_{q=0}^{L-1} q p_q(r_q) \quad (3)$$

2) Gray Level Co-occurrence Matrices (GLCM):

GLCM utilizes texture calculations in the second order. Measurement of textures in the first order applies statistical calculations, which is based on the pixel value of the original image, such as variance, and does not pay attention to neighbouring pixel relationships. In the second order, the relationship between the two pixel pairs of the original image is taken into account. GLCM employs five quantities, in the form of angular second moment (ASM), contrast, inverse different moment (IDM), entropy, and correlation.

ASM, which is a measure of image homogeneity is calculated in the following way:

$$ASM = \sum_{i=1}^{L}\sum_{j=1}^{L}(GLCM(i,j))^2 \quad (4)$$

In this case, L determines the level used for computing. Contrast, which is a measure of the contribution of variations in level of gray pixels, is calculated in the following way:

$$Contrast = \sum_{n=1}^{L} n^2 \left\{\sum_{|i-j|=n} GLCM(i,j)\right\} \quad (5)$$

The IDM feature is used to measure homogeneity. IDM is calculated as follows:

$$IDM = \sum_{i=1}^{L}\sum_{j=1}^{L}\frac{(GLCM(i,j))^2}{1+(i-j)^2} \quad (6)$$

Entropy states the size of gray level irregularities in the image. The value is high if the GLCM elements have relatively the same value. The value is low if the GLCM elements are close to 0 or 1. The formula for calculating entropy is:

$$Entropy - -\sum_{i=1}^{L}\sum_{j=1}^{L}(GLCM(i,j))\log(GLCM(i,j)) \quad (7)$$

Correlation which is a measure of linear dependency between gray levels in the image is calculated using the formula:

$$Correlation = \frac{\sum_{i=1}^{L}\sum_{j=1}^{L}(ij)(GLCM(i,j)) - \mu_i'\mu_j'}{\sigma_i'\sigma_j'} \quad (8)$$

where

$$\mu_i' = \sum_{i=1}^{L}\sum_{j=1}^{L} i * GLCM(i,j) \quad (9)$$

$$\mu_j' = \sum_{i=1}^{L} \sum_{j=1}^{L} j * GLCM(i,j) \tag{10}$$

$$\sigma_j^2 = \sum_{i=1}^{L} \sum_{j=1}^{L} GLCM(i,j)(i - \mu_i')^2 \tag{11}$$

$$\sigma_i^2 = \sum_{i=1}^{L} \sum_{j=1}^{L} GLCM(i,j)(i - \mu_i')^2 \tag{12}$$

3) Support Vector Machine (SVM)

SVM is a selection method that compares the standard parameters of a set of discrete values called candidate sets, and takes the one that has the best classification accuracy. SVM is one of the most influential and powerful tools for solving classifications[16]–[18]. SVM is a set of methods related to a learning method, for both classification and regression problems. With task-oriented, powerful, easy-to-do computational properties, SVM has achieved great success and is considered the current state-of-the-art classifier.

Two classes of data are described as circular and dotted points presented in this number. Intuitively observed, there are many hyperplanes decisions that can be used to separate the two data groups. However, what is depicted with this number is chosen as being advantageous in separating fields, because it contains a maximum margin between the two classes[19]. Therefore, in the goal of the SVM function, a regularization term represents an emergent margin. Especially as seen in this value, only those with full points are called support vectors mainly determining separating fields, while other points do not contribute to margins at all. In other words, only a number of important points for the classification of objectives within the SVM framework and as such must be taken.

The concept of SVM can be explained simply as an attempt to find the best hyperplane that has a function as a separator of two classes in the input space. For n-dimensional space, input data x_i (i = 1 ... k), which belongs to class 1 or class 2 and the associated label becomes -1 for class 1 and +1 for class 2. Figure 2 shows several patterns that are members of two classes: positive (denoted by +1) and negative (notated by –1). Patterns that are joined to negative classes are symbolized by squares, whereas patterns from positive classes are symbolized by circles. If the input data can be separated linearly, hyper plane separation can be given in the learning process in the classification problem, which is translated as an effort to find a line (hyperplane) that separates the two groups[19]. Various alternative discrimination boundaries are shown in Fig.2.

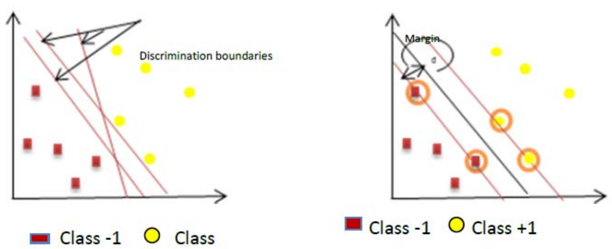

Fig. 2. SVM tries to find the best Hyperplane that separates both negative and positive classes

The best hyperplane separator between the two classes can be found by measuring the margin of the hyperplane and looking for the maximum point. Margin is the distance between the hyperplane and the closest data from each class. The closest subset of training datasets is called a support vector. The solid line in Figure 2 shows the best hyperplane, which is located right in the middle of the two classes, while the square and circle points marked in the black circle are support vectors. The effort to find the optimal hyperplane location is the core of the learning process in SVM.
Available data is denoted as x ∈ R d while each label is notated y_i ∈{-1+1} for i = 1,2,....,1 where l is the amount of data. It is assumed that both class-1 and +1 can be completely separated by a hyperplane, which has dimension of d.

IV. Design of Experiment

A. *Labeling of Data*

In the dataset the cropping process is carried out first with a size of 100 x 256. Fig.3 shows the data that has been cropped and labeled.

(a) (b)

Fig.3 (a) With Crack (b) No Crack

Data is classified into two groups, namely with crack and no crack.

B. *Segmentation based OTSU*

To obtain rough crack areas, the OTSU thresholding method is applied to attain global threshold values. Then the standard deviation of the filter response is calculated. To be conservative, the final threshold value is the sum of the OTSU threshold values and the half of the standard deviation. Fig.4 exhibits the rough crack area that was generated using the final threshold value.

(a) (b)

Fig.4 (a) Original (b) OTSU Segmentation

C. *Extraction Feature*

Feature extraction employs the GLCM method. This method utilizes 4 angles, which are 0^0, 45^0, 90^0 dan 135^0 angles. The results of feature extraction from one data can be seen from TABLE I. There are 40 image data used, so the number of data lines is 40 x 20 = 800 data lines.

D. Performance Test of SVM Model

Model performance testing in this study was carried out by comparing the existing kernel in SVM to obtain a model with the highest performance. The parameters applied in evaluating this kernel comparison are accuracy, precision, recall, ROC curve (AUC), and ANOVA statistical test[20]. The following will discuss the evaluation parameters that will be used as performance testing of the SVM model.

TABLE I. FEATURE EXTRACTION RESULTS OF THE FIRST DATA

	ASM	CONTRAST	IDM	ENTROPY	CORRELATION
0^0	3,37E+11	1,78E+17	0.1238	8.515.1	9,10E+11
45^0	2,37E+11	4,37E+17	0.084	8.85469	7,78E+11
90^0	2,58E+11	3,77E+17	0.0952	8.790.17	8,07E+11
135^0	2,15E+11	5,64E+17	0.0771	8.947.73	7,15E+11

1) Accuracy, precision, and recall

Accuracy can be defined as the level of closeness between the predicted value and the actual value. Precision shows the level of accuracy or precision in classification. Whereas recall serves to measure the actual positive proportions that are correctly identified. To measure accuracy, precise, and recall, confusion matrix is usually employed. Confusion matrix is a matrix measuring instrument applied to get the amount of class classification accuracy with the algorithm. The form of confusion matrix can be seen in TABLE II.

TABLE II. CONFUSION MATRIX OF TWO CLASSES

Confusion Matrix		**Real Value**	
		TRUE	**FALSE**
Prediction Value	**TRUE**	TP (True Positive) Correct result	FP (False Positive) Unexpected result
	FALSE	FN (False Negative) Missing result	TN (True Negative) Correct absence of result

In Table II the values of TP (true positive) and TN (true negative) indicate the level of classification accuracy. Generally, the higher the TP and TN values are obtained, the better the classification level of accuracy, precision, and recall are generated. If the predicted output label is true and the true value is false, then it is called false positive (FP). Whereas if the predicted output label is false and the true value is true then this is referred to as false negative (FN)[20]. The formulations for calculating accuracy, precision, and recall on the classification model formation are shown in Equation (13), Equation (14) and Equation (15).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} x100\% \quad (13)$$

$$Precision = \frac{TP}{TP+FP} x100\% \quad (14)$$

$$Recall = \frac{TP}{TP+FN} x100\% \quad (15)$$

2) ROC Curve

The ROC (receiver operating characteristic) curve is a measure to assess the ability of a classification system. This research employs ROC curve measurement tool to compare SVM kernels. The ROC curve was initially implemented in signal detection theory. Then it was developed and used in the fields of medicine, radiology, and other fields. ROC curves are often applied to assess classifications because they have the ability to evaluate algorithms quite well[20].

The ROC curve is a comparison graph between sensitivity (true positive rate (TPR)), which is translated into the vertical axis or y-axis coordinate, with specificity (false positive rate (FPR)), which is translated in the form of a curve. The formulations of sensitivity and specificity are presented in Equation (16), and Equation (17)[20].

$$Sensitifity = \frac{TP}{TP+FN} x100\% \quad (16)$$

$$Specificity = \frac{FP}{FP+TN} x100\% \quad (17)$$

AUC (area under curve) is the area under the ROC curve. The area of the AUC is always between values 0 to 1. The AUC is calculated based on the combined area of the trapezoid points (sensitivity and specificity).

The standard classification class tables based on the AUC values are depicted in TABLE III

TABLE III. CATEGORY OF CLASSIFICATION BASED ON AUC VALUE

AUC Value	**Classification Category**
0.90 - 1.00	Excellent
0.80 - 0.90	Good
0.70 - 0.80	Fair
0.60 - 0.70	Poor
0.50 - 0.60	Fail

V. RESULT AND DISCUSSION

In the study, training data is utilized to form classification models, namely the value of the extraction features of asphalt pavement crack image data. The performance of the SVM model applied to the five SVM kernel functions, which includes dot, radial, polynomial, neural, and anova kernels. Each experiment with various SVM models is then assesed using accuracy, precision, and recall values to get the best model.

TABLE IV exhibits a comparison of the accuracy of classifying data using SVM. The first assessment employs accuracy level. Seeing from TABLE IV, the highest accuracy level of 96.25% is obtained when applying the Anova kernel with C parameter of 0.5. Then, precision evaluator is utilized for the second assessment. TABLE V presents a comparison of the level of precision in classifying data using SVM in each kernel.

As seen from TABLE V, the greatest precision level of 96% is obtained when applying SVM with parameters C (penalties) of 500 and 1000 in the Dot kernel as well as in the Radial kernel with parameters C of 100, 500 and 1000.

The third assessment is to use the recall evaluator. TABLE VI presents a comparison of the recall rates of classifying data using SVM.

TABLE IV. ACCURACY LEVEL OF SVM CLASSIFICATION

Par C	Kernel Type				
	Dot	Radial	Polynomial	Neural	Anova
0.0	90.00	90	67.5	62.5	90
0.5	88.75	90	90	70	96.25
1	88.75	90	90	62.5	92.5
10	91.25	95	88.75	60	95
100	92.5	95	90	58.75	92.5
500	92.5	95	90	57.5	92.5
1000	92.5	95	90	58.75	92.5

TABLE V. PRECISION LEVEL OF SVM CLASSIFICATION

Par C	Kernel Type				
	Dot	Radial	Polynomial	Neural	Anova
0.0	93.5	93.5	71.31	62.67	93.5
0.5	93.5	93.5	92.67	71.67	94
1	93.5	93.5	92.67	62.67	89.33
10	95.5	94	90.67	58.67	95.5
100	95.5	96	91	57.17	93
500	96	96	90.17	57.17	93
1000	96	96	89	57.17	93

TABLE VI. RECALL LEVEL OF SVM CLASSIFICATION

Par C	Kernel Type				
	Dot	Radial	Polynomial	Neural	Anova
0.0	87.5	87.5	82.5	62.5	87.5
0.5	85	87.5	90	65	100
1	85	87.5	90	62.5	100
10	87.5	97.5	90	60	95
100	90	95	90	60	92.5
500	90	95	92.5	60	92.5
1000	90	95	92.5	60	92.5

The greatest recall level of 100% is achieved when applying the Anova kernel with parameter C (penalties) of 0.5 and 1. The next assessment is to measure the value of AUC (area under curve) on the ROC curve. TABLE VII depicts a comparison of the AUC level of data classification using SVM.

TABLE VII. AUC LEVEL OF SVM CLASSIFICATIONJ

Par C	Kernel Type				
	Dot	Radial	Polynomial	Neural	Anova
0.0	0.894	0.975	0.85	0.713	0.975
0.5	0.96	0.975	0.95	0.738	0.981
1	0.931	0.975	0.944	0.7113	0.981
10	0.969	0.969	0.956	0.631	0.956
100	0.956	0.944	0.925	0.5	0.95
500	0.963	0.944	0.944	0.438	0.95
1000	0.963	0.944	0.925	0.412	0.95

From TABLE VII, AUC (area under curve) values among the kernels are compared. A conclusion can be drawn that by selecting the right attributes (parameters) using the SVM method, it generally can increase the AUC (area under curve) value of the SVM model on several types of kernels tested, such as anova kernel.

The highest AUC (area under curve) level of 0.981% is obtained when applying the C (penalty) parameter of 0.5 and 0.1 using the Anova kernel.

Based on the SVM test results above, it can be concluded that SVM works relatively well with Anova kernel performance, because it provides the highest level of accuracy, precision and AUC compared to other kernels. Whereas the best recall value is produced by Dot and Radial kernels, because they have the largest value, compared to other kernels.

After all those assessments, the next step is to evaluate the best SVM classification model from different datasets. The SVM model with anova kernel and a penalty factor (C) parameter of 0.5 is used as a prediction reference.

The evaluation results of the model are then analyzed by matching the label prediction with the label crack and no crack, so that using confusion matrix, the accuracy, precision, and recall performance are obtained.

TABLE VIII shows the confusion matrix of the results, which are obtained by applying the SVM with Anova kernel and parameter of 0.5 to the asphalt crack detection data.

TABLE VIII. CONFUSION MATRIX OF SVM MODEL WITH ANOVA KERNEL AND PARAMETER C OF 0.5

	true crack	true no_crack	class precision
pred. crack	37	0	100.00%
pred. no_crack	3	40	93.02%
class recall	92.50%	100.00%	Accuracy: 96.25%

VI. CONCLUSION AND RECOMMENDATION

It can be concluded that the SVM method is able to perform an automatic computation for the detection and classification of asphalt pavement cracks. OTSU algorithm is implemented for the segmentation process, meanwhile GLCM method is utilized for feature extraction to earn features from the image datasets that have been collected.

Image processing techniques of segmentation and feature extraction are applied to achieve the most accurate prediction accuracy. The successful application of this approach has been demonstrated through the results of experiments and statistical tests of the SVM method. This approach can be used to study the mapping function between image input features and class output without crack and with crack. Based on the experimental results, the Anova kernel which is applied to SVM performs as the most competent classifier, because it produces the highest accuracy.

Therefore, the SVM algorithm combined with OTSU segmentation and GLCM 4 feature extraction is highly recommended for the classification of asphalt pavement cracks. With good prediction accuracy, the proposed model has demonstrated a high potential for implementation in asphalt pavement crack detection systems.

ACKNOWLEDGMENT

This paper is supported by USAID through Sustainable Higher Education Research Alliances (SHERA) Program - Centre for Collaborative (CCR) National Center for Sustainable Transportation Technology (NCSTT) with Grant No. IIE00000078-ITB-1.

REFERENCES

[1] A. Cubero-Fernandez, F. J. Rodriguez-Lozano, R. Villatoro, J. Olivares, and J. M. Palomares, "Efficient pavement crack detection and classification," *Eurasip J. Image Video Process.*, vol. 2017, no. 1, 2017.

[2] S. Li, Y. Cao, and H. Cai, "Automatic Pavement-Crack Detection and Segmentation Based on Steerable Matched Filtering and an Active Contour Model," *J. Comput. Civ. Eng.*, vol. 31, no. 5, pp. 1–9, 2017.

[3] M. Hao, C. Lu, G. Wang, and W. Wang, "An improved neuron segmentation model for crack detection-image segmentation model," *Cybern. Inf. Technol.*, vol. 17, no. 2, pp. 119–133, 2017.

[4] S. Wang and W. Tang, "Pavement crack segmentation algorithm based on local optimal threshold of cracks density distribution," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 6838 LNCS, pp. 298–302, 2011.

[5] S. C. Radopoulou and I. Brilakis, "Automated Detection of Multiple Pavement Defects," *J. Comput. Civ. Eng.*, vol. 31, no. 2, 2017.

[6] Y. C. J. Tsai and F. Li, "Critical Assessment of Detecting Asphalt Pavement Cracks under Different Lighting and Low Intensity Contrast Conditions Using Emerging 3D Laser Technology," *J. Transp. Eng.*, vol. 138, no. 5, pp. 649–656, 2012.

[7] C. Chen, "Evaluation of Iowa asphalt pavement joint cracking," 2014.

[8] Q. Zou, Y. Cao, Q. Li, Q. Mao, and S. Wang, "CrackTree: Automatic crack detection from pavement images," *Pattern Recognit. Lett.*, vol. 33, no. 3, pp. 227–238, 2012.

[9] E. Kim *et al.*, "Suppressed Instability of a-IGZO Thin-Film Transistors Under Negative Bias Illumination Stress Using the Distributed Bragg Reflectors," *IEEE Trans. Electron Devices*, vol. 63, no. 3, pp. 1066–1071, 2016.

[10] L. Sun, M. Kamaliardakani, and Y. Zhang, "Weighted Neighborhood Pixels Segmentation Method for Automated Detection of Cracks on Pavement Surface Images," *J. Comput. Civ. Eng.*, vol. 30, no. 2, pp. 1–11, 2016.

[11] A. Akagic, E. Buza, S. Omanovic, and A. Karabegovic, "Pavement crack detection using Otsu thresholding for image segmentation," *2018 41st Int. Conv. Inf. Commun. Technol. Electron. Microelectron. MIPRO 2018 - Proc.*, pp. 1092–1097, 2018.

[12] P. B. Prakoso and Y. Sari, "Vehicle detection using background subtraction and clustering algorithms," *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 17, no. 3. p. 1393, 2019.

[13] Y. Sari and P. B. Prakoso, "Detection of Moving Vehicle using Adaptive Threshold Algorithm in Varied Lighting," *2018 5th Int. Conf. Electr. Veh. Technol.*, pp. 136–141, 2018.

[14] P. Prasanna, K. Dana, N. Gucunski, and B. Basily, "Computer-vision based crack detection and analysis," *Sensors Smart Struct. Technol. Civil, Mech. Aerosp. Syst. 2012*, vol. 8345, p. 834542, 2012.

[15] N. Duc, H. Quoc, and L. Nguyen, "A novel method for asphalt pavement crack classification based on image processing and machine learning," *Eng. Comput.*, vol. 0, no. 0, p. 0, 2018.

[16] M. Gavilán *et al.*, "Adaptive road crack detection system by pavement classification," *Sensors*, vol. 11, no. 10, pp. 9628–9657, 2011.

[17] D. Ai, G. Jiang, L. Siew Kei, and C. Li, "Automatic Pixel-Level Pavement Crack Detection Using Information of Multi-Scale Neighborhoods," *IEEE Access*, vol. 6, pp. 24452–24463, 2018.

[18] A. G. C. Salvador Marques, "Automatic Road Pavement Crack Detection using SVM," 2012.

[19] R. Srinivas, "Managing Large Data Sets Using Support Vector Machines," pp. 1–93, 2010.

[20] W. Lefebvre-Ulrikson, G. Da Costa, L. Rigutti, and I. Blum, *Data Mining*. 2016.