Classification of Damaged Road Types Using Multiclass Support Vector Machine (SVM)

D R Sulistyaningrum¹, S A M Putri², B Setyono³, E Ahyudanari⁴, and D Oranova⁵

¹,²,³Department of Mathematics, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
⁴Department of Civil Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia
⁵Department of Informatics Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia

E-mail: dwiratna@matematika.its.ac.id

Abstract. Damaged roads had been disturbed by social activities and involved in traffic accidents. Identification and classification of the types of defected road are required to minimize its impact and before repairs. Digital image processing technology can identify and classify the type of damaged roads automatically. In this study, the classification of defected roads is automatic with a multiclass Support Vector Machine (SVM). There are three classes in the classification process, namely, alligators, potholes, and cracks. The process of recognizing defected roads uses a multiclass SVM classification model with polynomial and Gaussian kernel function and One Vs. All strategy and uses a cell size of 16 × 16 pixels during the Histogram of Oriented Gradients (HOG) feature extraction process. and produces an accuracy value of 78.85%.

1. Introduction

Roads are a land transportation infrastructure that has an essential role in supporting the economic, social, and cultural areas and various other aspects of the social community. Road conditions affect several social communities. Moreover, road conditions must be in the right circumstances to increase road users' comfort and safety. However, until now, there are still a lot of damaged roads. There is a problem caused by defected roads between prone to causing congestion, effected traffic accidents, and even casualties.

The damage roads survey has been carried out in two ways: manually and using a smart car to collect video data on-road situations. The survey was taken manually by the officer directly. They walked along the road, took pictures of road damage with cameras, and measured the damage level according to the type of road damage. The survey recorded it in the form report, and then handled it according to the type of damage to the road. This manual method requires more labor and takes a long time.

The image processing technology is currently developing rapidly in various fields. The civil engineering sector also utilizes digital image processing, one of which is to monitor road conditions. Digital image processing has several necessary steps to introduce a particular object. The process stages are feature extraction and classification. There are many kinds of feature extraction methods, one of which is the Histogram of Oriented Gradients (HOG). Rafal Kapela et al. used the HOG method in a
study entitled Asphalt-Surfaced Pavement Cracks Detection Based on Histograms of Oriented Gradients [1]. In this study, researchers discussed identifying types of damaged roads in the form of cracks using the HOG algorithm and obtained good identification results. The classification stage is carried out after the feature extraction stage. At this stage, object groups are grouped based on the object's features. In a study entitled Potholes Detection Based on Support Vector Machine (SVM) in the Pavement Distress Image by Jin Lin and Yayu Liu, Researchers used the SVM method to classify the pothole’s damage, and the classification results were good[2]. Muslim et al have studied how to count holes from video data[3]. The research conducted pothole recognition, then tracked and finally calculated the amount of damage.

In this study, we classified the types of defected roads using the Multiclass Support Vector Machine (Multiclass SVM) method, because the classification results were three classes. Types of damage roads that are the research object are data on potholes, cracks, and alligator. This study's results are expected to be an alternative in assessing road conditions by identifying the types of defected roads and followed by appropriate handling.

2. Method
Figure. 1 shows the proposed methods for the classification of the types of damage roads using SVM. It consists of several processing units: First, image acquisition. The Second is preprocessing to improve image quality. Third, feature extraction with HOG. The next is classifications using SVM. A detailed description of the scheme is described in the next discussion.

![Figure 1. The proposed classification method.](image-url)
2.1. Preprocessing
The preprocessing stage consists of two processes, namely resizing and gray scaling. The resizing image will have smaller dimensions, it makes the feature extraction process, and the classification process runs faster. The gray scaling process is carried out by changing the RGB input image to a grayscale image.

2.2. Feature extraction with Histogram of Oriented Gradient (HOG)
Features are individual attributes or properties used to identify an object. Histogram of Oriented Gradients (HOG) is a feature descriptor that is useful for extracting image features in shape-based object recognition. The following are the features extraction steps using HOG [4][5][6]:

a. Gamma or Square-Root and color normalized
This stage is often performed at the preprocessing stage of data. Gamma normalization can be calculated with \( \log (p) \). Meanwhile, calculating the root square normalization with \( \sqrt{p} \) of each pixel \( p \) in the input image. In color normalization, the RGB image will be processed into a grayscale image. Color normalization is useful for simplifying image gradient calculations.

b. Gradient computing
The gradient is the rate of change in local intensity at the position of a certain image pixel on the horizontal or vertical axis. In the gradient computation phase, the vertical gradient \( f_x(x,y) \) and horizontal gradient \( f_y(x,y) \) is calculated for the cell's pixel.

\[
\begin{align*}
f_x(x,y) &= f(x + 1,y) - f(x - 1,y) \\
f_y(x,y) &= f(x,y + 1) - f(x,y - 1)
\end{align*}
\]

After calculating the vertical and horizontal gradients, calculate the amount of gradient \( m(x,y) \) gradient orientation \( (\theta) \) for each pixel in the image.

\[
m(x,y) = \sqrt{f_x(x,y)^2 + f_y(x,y)^2}
\]

\[
\theta(x,y) = \tan^{-1}\left(\frac{f_y(x,y)}{f_x(x,y)}\right)
\]

c. Determine bin orientation
The Bins orientation represent gradient, at 0° – 180° for an "unsigned" gradient or 0° - 360° for a signed gradient. The determination of the number of bins can be determined freely. Naveet Dalal and Bill Trigs stated that unsigned gradients work better than signed gradients[5].

d. Block normalization
The following equation can denote block normalization.

\[
V_h = \frac{V_i}{\sqrt{\|V_i\|^2 + \epsilon^2}}
\]

where
- \( V_h \): normalized feature vector
- \( V_i \): overlapping feature vectors
- \( i = 1 \leq i \leq 36 \): feature vector index

\[
\|V\|_i = \sqrt{\sum_i V_i^2} \quad \|V\|_i^2 = V_1^2 + V_2^2 + \ldots + V_{36}^2
\]

\( \epsilon \): small constant to avoid zero division
2.3. Classification using SVM algorithm

Support Vector Machine is a popular machine learning algorithm used for classification based on kernels. SVM is a binary classification used to classify a dataset into two classes with a hyperplane. The hyperplane is good if this object can optimize the generalization bounds on a large enough sample. Margin is the maximum distance between support vectors in class 1 and class 2, located parallel to the hyperplane. The support vector is the data point closest to this object. The concept of SVM can be illustrated in Figure 3.

In the SVM classification, there are two types of input data: data that can be separated linearly (linearly separable data) and data that cannot be separated linearly (nonlinearly separable data). The following explains the differences in resolution between data that can be separated linearly or not [8][9].

a. Linearly separable data

The linearly separated data is denoted as follows:

\[ x_i = \{x_1, x_2, ..., x_n\} \in \mathbb{R}^d \]

where \( x_i \) : input data, \( i = 1, 2, ..., N \).

\( N \) : the amount of input data

\( x_1, x_2, ..., x_n \) : features

\( n \) : many features

\( \mathbb{R}^d \) : real number to the power of \( d \)

Before classification, each data is labelled according to its characteristics. The dataset in class 1 is labelled (+1), while class 2 is labeled (-1). Class label or target is denoted as \( y_i \in \{-1, +1\} \).
Therefore, the data used is a pair \((x_1, y_1), \ldots, (x_N, y_N) \in X \times \{-1, +1\}\). The determination of the hyperplane line on linear separated data can be denoted as the following equation:

\[ x_i \cdot w + b = 0 \]

where
- \(w\) : load vector parameter
- \(b\) : bias parameters
- \(x_i\) : input data

The largest margin is obtained by maximizing the distance value between the hyperplane and support vector with the formula \(\frac{1}{||w||}\). With the quadratic programming problem, the hyperplane optimization problem can be solved by finding the minimum point with the following equation:

\[
\min \frac{1}{2} ||w||^2 \\
\text{with constraint} \\
y_i(x_i \cdot w + b) - 1 \geq 0, \forall i = \{1, 2, \ldots, N\}
\]

where
- \(||w||\) : Norm Euclidian vector \(w\)
- \(w\) : load vector parameter
- \(y_i\) : target class
- \(x_i\) : input data
- \(b\) : bias parameters

The result of minimizing the norm vector \(w\) is computed by minimizing the Lagrange Multipliers so that equation six is obtained.

\[
\min L_p(w, b, \alpha) \equiv \frac{1}{2} ||w||^2 - \sum_{i=1}^{N} \alpha_i (y_i(x_i \cdot w + b)) + \sum_{i=1}^{N} \alpha_i \\
\text{where} \\
L_p(w, b, \alpha) \quad \text{: Lagrange Multiplier with} \ w, b, \alpha \\\n\min L_p (w, b, \alpha) \quad \text{: Minimum Lagrange Multiplier with} \ w, b, \alpha \\
\alpha_i \quad \text{: Lagrange coefficient (} \alpha_i \geq 0, \ i = 1, 2, \ldots, l) \\
\]

To minimize \(L_p\) against \(w\) and \(b\) is necessary to minimize the load vector \((w)\) and bias \((b)\) and maximize the Lagrange coefficient \((\alpha_i)\). The solution used in this problem is the Karush-Kuhn-Tucker (KKT) condition approach. The calculation of the KKT condition is Equation 7 and Equation 8. The result of calculating Equation 7 is Equation 9, while the result of calculating Equation 8 is Equation 10.

\[
\frac{\partial}{\partial b} L_p(w, b, \alpha) = 0 \\
\frac{\partial}{\partial w} L_p(w, b, \alpha) = 0 \\
\sum_{i=1}^{N} \alpha_i y_i = 0 \\
w = \sum_{i=1}^{N} \alpha_i y_i x_i 
\]

where
- \(\frac{\partial}{\partial b} L_p(w, b, \alpha)\) : the partial derivative of \(L_p\) for \(b\)
- \(\frac{\partial}{\partial w} L_p(w, b, \alpha)\) : the partial derivative of \(L_p\) for \(w\)

Equations 9 and 10 are substituted for Equation 6, resulting in a dual problem with the following equation.

Maximum:

\[
\sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^{N} \alpha_i y_i x_i x_j \\
\text{with constraint}
\]
\[ \sum_{i=1}^{N} \alpha_i y_i = 0, \alpha_i \geq 0 \]

The result of this equation is the value \( \alpha_i \), which is used to find \( w \). Training data with a value of \( \alpha_i \geq 0 \) is a support vector, while training data with a value of \( \alpha_i = 0 \) is not a support vector. After finding the solution to the quadratic programming \((\alpha_i)\) problem, the class of test data denoted as \( x \) can be determined based on the value of the decision function in the following equation.

\[ f(x_i) = \text{sign} \left( \sum_{i=1}^{ns} \alpha_i y_i x_i x_d + b \right) \]

where  
- \( x_i \): input data to be classified  
- \( x_d \): support vector  
- \( ns \): the number of support vectors

b. Nonlinearly separable data

In the SVM classification, not all input data used can be separated linearly. To solve this problem is necessary to modify Equation 10 by adding a slack variable \((\xi_i)\) or often referred to as a soft margin hyperplane. So that Equation 10 changes to the following equation.

\[ \min \frac{1}{2} \| w \|^2 + C \left( \sum_{i=1}^{N} \xi_i \right) \]

with constraint

\[ y_i (x_i \cdot w + b) \geq 1 - \xi_i, \text{ for } \xi_i \geq 0, \ i = 1, 2, \ldots, N \]

where  
- \( \xi_i \): variable slack or classification error  
- \( C \): penalty parameter  
- \( w \): load vector parameter  
- \( b \): bias parameters  
- \( x_i \): input data  
- \( y_i \): target class

In the case of this soft margin hyperplane, minimizes not only the load vector \((w)\) and bias \((b)\) but also minimizes parameter \( C \). The parameter \( C \) is chosen to control the tradeoff between margin and classification error \((\xi_i)\).

c. Kernel functions

The kernel function in this SVM classification is useful for separating data in certain classes based on the kernel. Figure 4. shows the transformation from input space to feature space with mapping function.

\[ K(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j) \]

Each data in the input space is mapped into a feature space (new vector space) using the mapping function \( \varphi(.) \). The kernel function \( K(x_i, x_j) \) can be formulated into the following equation.
with $K(x_i, x_j)$ is kernel function, and $\varphi(x_i)$ is the mapping function.

The kernel function provides convenience to the SVM learning process[10]. Commonly used kernel functions are as follows:

i. Linear kernel
   
   
   $K(x_1, x_2) = x_1^T x_2$

ii. Polynomial

   
   $K(x_1, x_2) = (x_1^T x_2 + 1)^p$

iii. Radial Basis Function (RBF) or Gaussian

   
   $K(x_1, x_2) = \exp\left(\frac{-|x_1 - x_2|^2}{2\sigma^2}\right)$

with $\sigma$ is the independent parameter, and $p$ is the order of the polynomial kernel.

\[ \text{d. Multiclass Support Vector Machine} \]

The SVM concept is a classification of 2 classes. However, to solve more than two classes’ classification problem, a multiclass SVM algorithm was developed. Two multiclass SVM approaches are most commonly applied, namely One Vs. All and One Vs. One[7][11]. Figure 5. shows illustration of One Vs. All Four Classification.

\[ \text{e. Program Performance Evaluation} \]

The system performance evaluation is calculated by measuring the value of accuracy, precision, and recall. The value of accuracy, precision, and recall can be calculated with the following formula [12]:

\[
\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{11}
\]

\[
\text{Precision}_i = \frac{TP_i}{TP_i + FP_i} \tag{12}
\]

\[
\text{Recall} = \frac{TP_i}{TP_i + FN_i} \tag{13}
\]

Where  
TP: True Positive, namely the number of positive data classified correctly

TN: True Negative, which is the amount of negative data classified correctly

FN: False Negative, which is the number of negative data but is classified incorrectly

FP: False Positive, namely the number of positive data but classified incorrectly

Figure 5. (a) Illustration of one vs. all classifications for four classes, (b) Illustration of one vs. one classification for four classes.
3. Experiments design

This study aims to classify the types of road damage based on road image. In this study, we used 260 images of road damage as experimental data. These images are obtained from research[12]. The input data used for training data is a single road damage image snippet. The road damage image examples is showed in Figure 6.

![Image of road damage examples](image1.png)

Figure 6. Example of road damage image dataset, (a) alligator, (b) Pothole, (c) Crack [12].

There are two processes carried out in this classification process, namely:

a. The training process is the process of training a system with input data that has been previously processed to recognize when given new input. Figure 7. shows illustration of training process.

b. The testing process is the decision making the process of the test image. Figure 8. Illustration of Testing Process. Figure 8. shows illustration of testing process.

![Illustration of Training Process](image2.png)

Figure 7. Illustration of training process.
4. Result and discussion

The number of test data is in the form of pieces of single damage road images, namely 52 pieces. The types of it were alligators, potholes and cracks. In the previous image preprocessing stage, all image data were normalized to the image size to $128 \times 64$. This is done to equalize the number of features extracted from each data. In this study, cell size parameters were selected in an effort to obtain the best classification model. The selected cell size parameters are $8 \times 8$ pixels, $16 \times 16$ pixels, and $32 \times 32$ pixels.

At this classification stage, it is taken using two SVM strategies and three kernel functions. The multilevel SVM strategy used in this study is One Vs. One (OVO) and One Vs. All (OVA). While the kernel functions used in this research are: Linear, Polynomial, and Gaussian. The test scenarios for the SVM classification model and evaluation results are shown in Tables 1, 2 and 3 for each kernel function.
Table 1. Result of classification scenario with linear kernel.

| No | Strategy | Cell Size | Accuracy | Precision | Recall | Time of Training | Time of Testing |
|----|----------|-----------|----------|-----------|--------|-----------------|----------------|
| 1. | OVO      | 8 × 8     | 63.46%   | 43.75%    | 63.64% | 0.080172        | 0.010802       |
|    |          |           |          | 70.00%    | 58.33% |                 |                |
|    |          |           |          | 75.00%    | 70.59% |                 |                |
| 2. | OVO      | 16 × 16   | 71.15%   | 62.50%    | 66.67% | 0.130378        | 0.027513       |
|    |          |           |          | 80.00%    | 69.57% |                 |                |
|    |          |           |          | 68.75%    | 78.57% |                 |                |
| 3. | OVO      | 32 × 32   | 40.38%   | 6.25%     | 100%   | 0.079922        | 0.005088       |
|    |          |           |          | 100%      | 39.22% |                 |                |
|    |          |           |          | 0%        | NaN    |                 |                |
| 4. | OVA      | 8 × 8     | 67.31%   | 50.00%    | 72.73% | 0.081510        | 0.014872       |
|    |          |           |          | 70.00%    | 60.87% |                 |                |
|    |          |           |          | 81.25%    | 72.22% |                 |                |
| 5. | OVA      | 16 × 16   | 76.92%   | 62.50%    | 83.33% | 0.093092        | 0.008970       |
|    |          |           |          | 75.00%    | 68.18% |                 |                |
|    |          |           |          | 93.75%    | 83.33% |                 |                |
| 6. | OVA      | 32 × 32   | 57.69%   | 25.00%    | 50.00% | 0.063560        | 0.006458       |
|    |          |           |          | 95.00%    | 52.78% |                 |                |
|    |          |           |          | 43.75%    | 87.50% |                 |                |

Table 2. Result of classification scenario with polynomial kernel.

| No | Strategy | Cell Size | Accuracy | Precision | Recall | Time of Training | Time of Testing |
|----|----------|-----------|----------|-----------|--------|-----------------|----------------|
| 1. | OVO      | 8 × 8     | 71.15%   | 62.50%    | 76.92% | 0.062018        | 0.011717       |
|    |          |           |          | 75.00%    | 65.22% |                 |                |
|    |          |           |          | 75.00%    | 75.00% |                 |                |
| 2. | OVO      | 16 × 16   | 75.00%   | 75.00%    | 70.59% | 0.080729        | 0.007063       |
|    |          |           |          | 80.00%    | 76.19% |                 |                |
|    |          |           |          | 68.75%    | 78.57% |                 |                |
| 3. | OVO      | 32 × 32   | 69.23%   | 50.00%    | 61.54% | 0.069003        | 0.005305       |
|    |          |           |          | 85.00%    | 70.83% |                 |                |
|    |          |           |          | 68.75%    | 73.33% |                 |                |
| 4. | OVA      | 8 × 8     | 71.15%   | 62.50%    | 76.92% | 0.081510        | 0.014872       |
|    |          |           |          | 70.00%    | 66.67% |                 |                |
|    |          |           |          | 81.25%    | 72.22% |                 |                |
| 5. | OVA      | 16 × 16   | 78.85%   | 75.00%    | 80.00% | 0.069796        | 0.013598       |
|    |          |           |          | 80.00%    | 76.19% |                 |                |
|    |          |           |          | 81.25%    | 81.25% |                 |                |
| 6. | OVA      | 32 × 32   | 59.62%   | 43.75%    | 50.00% | 0.088578        | 0.005657       |
|    |          |           |          | 75.00%    | 62.50% |                 |                |
|    |          |           |          | 56.25%    | 64.29% |                 |                |
Table 3. Result of classification scenario with Gaussian kernel.

| No | Strategy | Cell Size | Accuracy | Precision | Recall | Time of Training | Time of Testing |
|----|----------|-----------|----------|-----------|--------|-----------------|----------------|
| 1. | OVO      | 8 × 8     | 38.46%   | 0%        | NaN    | 0.071118        | 0.013343       |
| 2. | OVO      | 16 × 16   | 75.00%   | 75.00%    | 75.00% | 0.073310        | 0.007195       |
| 3. | OVO      | 32 × 32   | 48.08%   | 95.00%    | 42.22% | 0.064938        | 0.005305       |
| 4. | OVA      | 8 × 8     | 38.46%   | 100%      | 38.46% | 0.067877        | 0.014383       |
| 5. | OVA      | 16 × 16   | 78.85%   | 80.00%    | 72.73% | 0.081800        | 0.009436       |
| 6. | OVA      | 32 × 32   | 65.38%   | 43.75%    | 58.33% | 0.061463        | 0.006837       |

Based on the simulation results, it is known that the best model produced based on each kernel is in scenario number 5. The parameters used in scenario number 5 are SVM strategy with OVA level with HOG cell size 16 × 16. And the best kernel are polynomial and Gaussian.

5. Conclusion

Based on experiments and discussions on the test results that have been carried, then some conclusions can be drawn as follows:

1. This study has succeeded in classification of damaged road using the Histogram of Oriented Gradients and SVM method, with the stages of the process namely pre-processing, feature extraction with HOG, and the classification process with Multiclass SVM.
2. The highest accuracy value at the time of testing was 78.85%. The accuracy value are obtained using a classification model with Polynomial and Gaussian kernel function, the SVM strategy with One Vs. All levels, and the HOG cell size of 16 × 16 pixels.

Acknowledgments

This research is supported by the Institute for Research and Community Development (Institute for Research and Community Development / DRPM) Institut Teknologi Sepuluh Nopember Surabaya with Funding Agreement Letter number 1372 / PKS / ITS / 2020.

References

[1] Kapela R, Sniatala P, Turkot A, Rybarczyk A, Pozarycki A, Rydzewski P, Wyczalek M and Bloch A 2015 Asphalt surfaced pavement cracks detection based on histograms of oriented gradients Proc. 22nd Int. Conf. Mix. Des. Integr. Circuits Syst. Mix. 2015 579–84
[2] Lin J and Liu Y 2010 Potholes Detection Based on SVM in the Pavement Distress Image 2010 Ninth International Symposium on Distributed Computing and Applications to Business, Engineering and Science pp 544–7
[3] Muslim M, Sulistyaningrum D R and Setiyono B 2020 Detection and counting potholes using
morphological method from road video AIP Conf. Proc. 2242 30011

[4] Sulistyaningrum D R, Ummah T, Setiyyono B, Utomo D B, Soetrisno and Sanjoyo B A 2020 Vehicle detection using histogram of oriented gradients and real adaboost J. Phys. Conf. Ser. 1490

[5] Dalal N and Triggs B 2005 Histograms of oriented gradients for human detection 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05) vol 1 pp 886–93 vol. 1

[6] Nixon M S and Aguado A S 2002 5 - Feature extraction by shape matching ed M S Nixon and A S B T-F E and I P Aguado (Oxford: Newnes) pp 161–216

[7] Sulistyaningrum D R, Rasyida A and Setiyono B 2020 Rice disease classification based on leaf image using multilevel Support Vector Machine (SVM) J. Phys. Conf. Ser. 1490

[8] Chamasemani F F and Singh Y P 2011 Multi-class Support Vector Machine (SVM) Classifiers - An Application in Hypothyroid Detection and Classification 2011 Sixth International Conference on Bio-Inspired Computing: Theories and Applications pp 351–6

[9] Cusano C, Ciocca G and Schettini R 2003 Image annotation using SVM Internet Imaging V 5304 330–8

[10] Diani R 2017 Analisis Pengaruh Kernel Support Vector Machine (SVM) pada Klasifikasi Data Microarray untuk Deteksi Kanker Indones. J. Comput. 2 109

[11] Duan K, Rajapakse J and Nguyen M N 2007 One-Versus-One and One-Versus-All Multiclass SVM-RFE for Gene Selection in Cancer Classification EvoBIO

[12] Sulistyaningrum D R, Oranova D, Setiyono B, Ahystanari E, Pramestya R H and Muklash I 2019 Pavement Distress Classification Using Deep Learning Method Based on Digital Image Proceedings series : Advances in Engineering Research, Proceedings of the 2nd International Symposium on Transportation Studies in Developing Countries (ISTSDC 2019)