Eduvest – Journal of Universal Studies
Volume 5 Number 9, September, 2025
p- ISSN 2775-3735- e-ISSN 2775-3727

Classify a path on tire by using Logistic Regression and Support Vector Machine (SVM)Based on VGG-16, VGG-19, and INCEPTION V3 Modes

Sukma Sufryanto[1], Imam Yuadi[2]
Airlangga University, Surabaya, Indonesia
Email : sukma.sufryanto-2024@pasca.unair.ac.id[1], imam.yuadi@fisip.unair.ac.id[2]

ABSTRACT

This study focuses on the classification of tire tread patterns using machine learning and deep learning approaches, emphasizing Logistic Regression (LR) and Support Vector Machine (SVM) combined with feature extraction methods like Inception V3, VGG-16, and VGG-19. Results indicate that Inception V3 outperformed other feature extraction methods, yielding the highest classification accuracy (CA) of 93.2% when used with SVM. SVM demonstrated superior robustness and adaptability, especially in handling complex data, as evidenced by its high AUC values (up to 0.987) across multiple configurations. Logistic Regression, while slightly less robust, performed consistently well with simpler features, achieving stable metrics with VGG-16 (AUC: 0.976, CA: 90.7%). These findings highlight the importance of selecting appropriate feature extraction and classification combinations to optimize performance. The study recommends using Inception V3 with SVM for high-accuracy applications and Logistic Regression for scenarios prioritizing computational efficiency. These insights contribute to developing adaptive and efficient tire classification systems suitable for diverse road and environmental conditions.

KEYWORDS *Interest, Facilities and Infrastructure, Instructor Competence, Learning Methods, Certification Implementation, and Certification Benefits.*

INTRODUCTION

This study focuses on the classification of tire tread patterns using machine learning and deep learning approaches, emphasizing *Logistic Regression* (LR) and *Support Vector Machine* (SVM) combined with feature extraction methods like *Inception V3*, *VGG-16*, and *VGG-19*. A tire is a device that covers the wheels of a vehicle and plays a crucial role in reducing vibrations caused by road surface irregularities, protecting wheels from wear and damage, and providing stability between the vehicle and the ground to enhance acceleration and facilitate movement (A.P et al., 2022; Barbosa & Magalhães, 2015; Dong et al., 2017; Li et al., 2018; Liu et al., 2019; Zhang et al., 2022). Tires are essential components for driving safety and performance, as correct tire selection improves driver safety by preventing slides, reducing vehicle operating costs, enhancing performance, and simplifying maintenance. Each tire's tread is specifically designed to suit particular road conditions, providing optimal traction on both dry and wet surfaces. According to Smith et al. (2020), tire tread pattern directly affects vehicle stability and fuel efficiency. Tire tread pattern classification is critical for performance analysis, particularly for detecting wear or mismatches that can increase accident risk. For example, Hasegawa et al. (2010) demonstrated that optimal tread pattern design reduces aquaplaning risk by efficiently channeling water from the tire-road contact area, and Saka et al. (2012) discussed how tire geometry and tread depth influence traction on various surfaces such as wet asphalt, gravel, and mud.

With advances in artificial intelligence, machine learning methods are increasingly applied to image processing for automatic tire tread pattern classification. Convolutional Neural Networks (CNNs) have become popular for image classification in the automotive industry: Krizhevsky et al. (2017) noted that CNNs extract important image features more effectively than traditional methods. CNN models such as *VGG-16*, *VGG-19*, and *Inception V3* have demonstrated effectiveness in pattern recognition across various industries. *VGG-16* and *VGG-19* offer layered architectures enabling highly accurate feature mapping, while *Inception V3* provides computational efficiency through factorized convolutions. Besides CNNs, classification algorithms like *Logistic Regression* and *Support Vector Machine* (*SVM*) are widely used in image processing. *Logistic Regression* is a statistical method commonly used for binary or multi-class classification, based on probability to categorize objects. Meanwhile, as described by Cortes & Vapnik (1995), *SVM* effectively handles high-dimensional data and achieves optimal class separation via hyperplanes. Both methods have been applied in pattern recognition and image analysis in the vehicle industry to detect tire tread wear and optimize tread design, contributing to driving safety.

Combining deep learning with machine learning is a growing approach in object classification, including tire tread analysis. According to LeCun et al. (2015), deep learning enables high-level feature extraction, whereas machine learning algorithms such as *Logistic Regression* and *SVM* perform superior classification using these extracted features. Thus, integrating CNNs as feature extractors with *SVM* and *Logistic Regression* classifiers enhances accuracy and efficiency. This study evaluates the performance of such combined models in automatically classifying tire lane patterns using images from different road conditions.

Features extracted from *VGG-16*, *VGG-19*, and *Inception V3* models serve as inputs for *Logistic Regression* and *SVM* algorithms to classify tire tracks. He et al. (2019) found that CNN feature extraction improves classification accuracy by generating more representative image features than traditional methods. By experimenting on tire image datasets, this research assesses the accuracy and effectiveness of each model combination and compares the performance of *Logistic Regression* versus *SVM* in tire track classification.

This research aims to develop an automatic classification system to assist the automotive industry in identifying and analyzing tire tread patterns quickly and efficiently. Given the rising demand for predictive vehicle maintenance technologies, the study contributes to AI-based tire analysis methods. It evaluates not only classification accuracy but also processing speed and efficiency on large datasets, offering a reference for advanced, automated tire inspection systems in the automotive sector.

Related research shows that Convolutional Neural Networks (CNNs) are often applied for feature extraction from tire images, followed by classification using *SVM*, *Logistic Regression*, or Random Forest methods. Krizhevsky et al. (2017) highlighted CNNs' superiority in image processing due to automatic high-level feature extraction without manual feature engineering. CNN applications in automotive contexts include tire wear detection, brand identification, and groove classification. *VGG-16*, *VGG-19*, and *Inception V3* are favored CNN models due to their deep architectures capturing complex image patterns.

Studies comparing CNNs to traditional machine learning classifiers for tire pattern recognition (He et al. 2019; Wang et al. 2021) reaffirm that combining CNN feature extraction with *SVM* typically improves accuracy, especially with complex, high-dimensional features.

Conversely, *Logistic Regression* remains competitive when features are simpler, offering faster computation and easier interpretation. Smith et al. (2020) noted that while *VGG* models yield high accuracy, *Inception V3* provides better computational speed without significant accuracy loss, hence this study's focus on these models.

Support Vector Machine (Cortes & Vapnik, 1995) is praised for its ability to handle high-dimensional data and optimize decision boundaries, enhancing image classification accuracy across domains including medical images and handwriting recognition. In tire groove classification, *SVM* improves accuracy compared to purely neural network-based methods.

Logistic Regression also plays a role in probabilistic pattern analysis. Hosmer & Lemeshow (2000) describe its use in multi-category classification based on predictor variables. While performing slightly worse than *SVM* or CNN classifiers in some tire studies (Li et al. 2018), it remains valuable for rapid processing and straightforward results interpretation.

CNN-based techniques extend to related applications such as road pattern recognition and tire characteristic classification. Zhao et al. (2019) utilized CNN combined with morphology-based segmentation to differentiate tire groove textures and shapes, which traditional manual methods struggle to capture.

Dataset quality and image preprocessing critically influence classification outcomes. Zhang et al. (2020) reported that augmentation techniques like rotation, flipping, and contrast adjustment can boost classification accuracy by up to 10%, important given the variable orientations and lighting in tire images. This study adopts normalization, data augmentation, and contrast enhancement to improve groove pattern classification performance.

Key findings of this study reveal that the combination of *Inception V3* and *SVM* achieves the highest accuracy (93.2%) and AUC (0.987), underscoring *SVM*'s advantages with complex data. Meanwhile, *Logistic Regression* combined with *VGG-16* remains favorable for computational efficiency. The study's dataset and preprocessing methods tailor to real-world scenarios, delivering practical AI solutions for automated tire inspection.

Overall, combining CNN feature extraction with machine learning classifiers like *SVM* and *Logistic Regression* shows strong potential in tire groove classification. Prior research confirms CNNs extract superior features, while *SVM* and *Logistic Regression* yield more optimal classification than purely neural network classifiers. This study integrates these strengths for vehicle tire groove pattern classification, leveraging optimally processed datasets to enhance accuracy and efficiency. In doing so, it contributes academically to AI in automotive research and offers industry-ready, innovative AI-based tire analysis methods, with model selection and performance analyses serving as valuable guides for developing advanced tire tread pattern classification systems.

RESEARCH METHODS

A. Equipment used

To analyze the objects in this study using the following tools:

1. Internet
2. laptop with specifications

Figure 1. Illustration of the equipment used
Source: Research document

B. Object Capture

Images are selected and taken through the internet browsing process with the help of google images. The image capture process is done by screencapturing the object to be classified. The objects taken are images of tire grooves which are grouped into 3 (three) parts, namely: symmetrical grooves, asymmetrical grooves, and directional grooves.

C. Tire Groove Type/Pattern

There are 3 (three) types of tire groove types that are classified in this article, which can be seen in the following figure:

No.	Object	Description
1.		Symmetrical groove tires are one type of tread groove that is often used in passenger cars. In symmetrical grooves, they generally have the same development, either on the side treads or in the center.
2.		Asymmetrical tires, designed with different tread sizes between the side tread and the inner tread. This type of tire has the advantage of being used in all types of weather, be it during dry or rainy road situations. No wonder this type of tire is often referred to as an All Seasons tire.
3.		This type of tire has a groove on the tread that is unidirectional, meaning that all four tires have the same groove direction, either towards the front or back. Tires with this type of groove generally have a tread shape like the letter V.

Figure 2. Tire groove/pattern type
Source: Research document

D. Research Flow

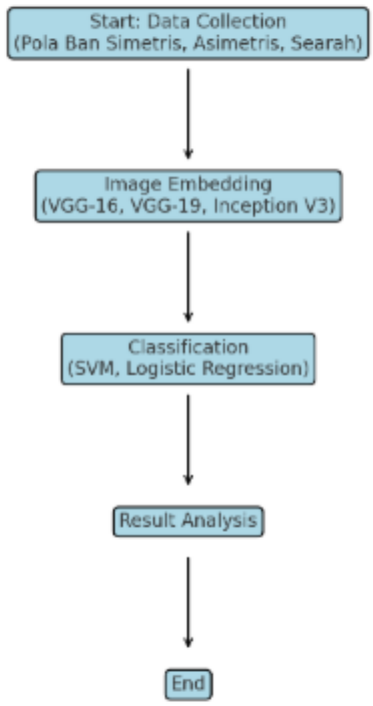


Figure 3. Research Flowchart
Source: Research document

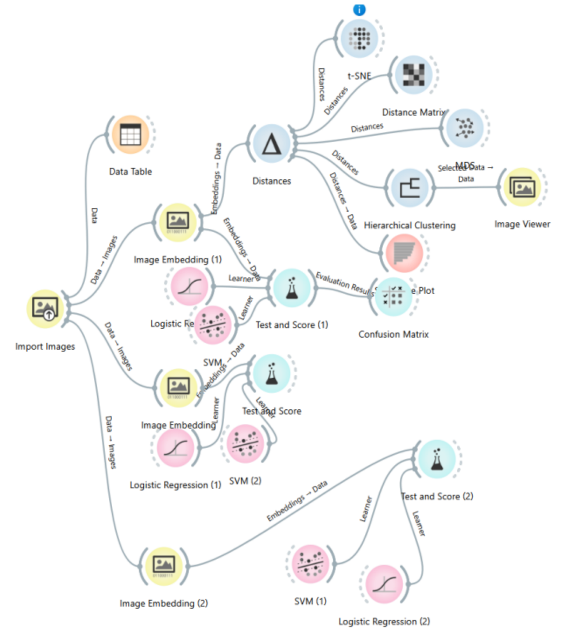
1. Data Collection (Symmetrical, Asymmetrical, Unidirectional Tire Pattern)
   The first step was to collect data on tire pattern images from three main categories: symmetrical, asymmetrical, and unidirectional. Symmetrical patterns have identical designs on both sides of the center, asymmetrical patterns differ between the inner and outer sides, while unidirectional patterns are designed to move in one specific direction. The data will be collected through capturing images using screen capture and categorized based on the type of pattern. This step includes data cleaning to ensure accuracy, such as removing duplicate or irrelevant images. The resulting dataset should be representative and of high quality to support subsequent analysis.
2. Image Embedding (VGG-16, VGG-19, Inception V3)
   At this stage, the important features of the tire pattern image are extracted using deep learning models and in this artiker using 3 types namely VGG-16, VGG-19, and Inception V3. These models help convert the image into a numerical representation (embedding) that reflects the visual characteristics of the image. The process starts with preprocessing, such as resizing the image to fit the model input, pixel normalization, and data augmentation if needed. The image is then processed by the model to produce an embedding at a specific layer, usually the final layer before the output. This representation becomes the basis for the next stage of classification.
3. Classification (SVM, Logistic Regression)
   The next step after the images have been extracted is to classify the tire patterns using machine learning algorithms, namely Support Vector Machine (SVM) and Logistic Regression. SVM serves to separate the data with maximum margin for proper classification, while Logistic Regression is used to model the relationship between features and target categories. The data is divided into training and test data to train the model and measure its performance. The model categorizes tire patterns into classes such as optimal, moderate, or poor. The classification results are an initial indicator to assess the effectiveness of the model on the dataset.
4. Result Analysis
   Evaluating the classification results to assess the performance of the model is an activity at this stage. The analysis is done by comparing the model predictions with the actual labels and calculating evaluation metrics such as accuracy, precision, recall, and F1-score. The evaluation results are used to identify the strengths and weaknesses of the model, such as bias towards certain classes or difficulty in recognizing certain patterns. If the model performance is not optimal, improvements are made, for example through hyperparameter tuning or data addition. This stage ensures that the developed system can meet the specific needs of the application.
5. End

The last step is to summarize the results of the analysis and provide recommendations for the next steps. The conclusion contains an assessment of the accuracy of the model, the most effective algorithm implementation, and the necessary improvement steps. In addition, these results can be applied in real product development, such as the design of more optimal tire patterns for specific road conditions. By taking a systematic approach, this stage ensures that the research produces solutions that are relevant and practically applicable.

RESULT AND DISCUSSION

a. Classification Model

The classification process is carried out using orange soft software, with import data as many as 118 images with 3 types of tire grooves which can be seen in Figure 4.



Pigure 4. Classification model with logistic regression and support vector machine
Source: The results of the Orange soft analysis

The Image embedding process is carried out by classifying the 118 that have been imported into the orange.Embedder application is used by using 3 (three) modes namely: Inception V3, VGG-16 and VGG-19. Furthermore, to see the distribution of data, you can see Figure 5, namely this diagram can be used for cluster or distribution analysis in a dataset with three different classifications.

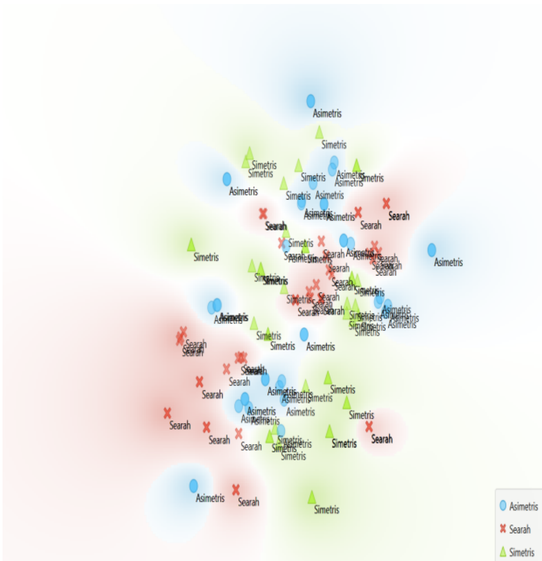


Figure 5. Result of Feature Extraction
Source: The results of the Orange soft analysis

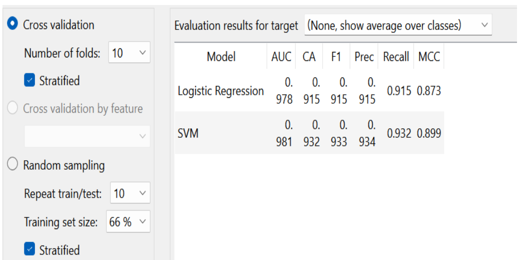
b. Logistic Regression Model

The learning models used are Logistic Regression and SVM. Logistic Regression uses a sigmoid function to convert the model output into a probability value between 0 and 1. The sigmoid function has the formula σ(z)=11+e−z\sigma(z) = \frac{1}{1 + e^{-z}}σ(z)=1+e−z1, where zzz is a linear combination of input features and weights (z=wTx+bz = w^T x + bz=wTx+b).

W : Weight Vector
X : Input Feature Vector
b : Bias

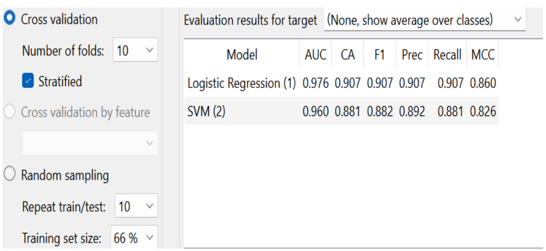
The probabilities generated from the sigmoid can be converted into class predictions by setting a certain threshold, usually 0.5. In the context of this dataset, Logistic Regression performed well with metrics such as AUC of 0.978 and recall of 0.915, indicating that the model has an excellent ability to identify positive classes.

Support Vector Machine (SVM) is a machine learning algorithm used for classification and regression tasks by finding the best hyperplane that separates the classes in a dataset. SVM maximizes the margin between the separating hyperplane and the closest data from each class, called support vectors. The algorithm can work linearly or non-linearly, depending on the use of kernels such as linear, polynomial, or radial basis function (RBF). RBF kernels, for example, allow SVMs to map data to a higher dimensional space to handle data that cannot be linearly separated. In this evaluation, SVM showed superior performance with an AUC of 0.981, classification accuracy of 93.2%, and recall of 93.2%, demonstrating its excellent ability to separate classes and provide accurate predictions even under complex data conditions. The data can be seen in Figure 6, Figure 7 and Figure 8.



Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression	0. 978	0. 915	0. 915	0. 915	0.915	0.873
SVM	0. 981	0. 932	0. 933	0. 934	0.932	0.899

Figure 6. Result Test and score type Logistic Regression and SVM Inception V3
Source: Hasil Inception V3 (AUC 0.987)



Model	AUC	CA	F1	Prec	Recall	MCC
Logistic Regression (1)	0.976	0.907	0.907	0.907	0.907	0.860
SVM (2)	0.960	0.881	0.882	0.892	0.881	0.826

Figure 7. Result Test and score type Logistic Regression and SVM VGG-16
Result: Result of VGG-16 (AUC 0.976)

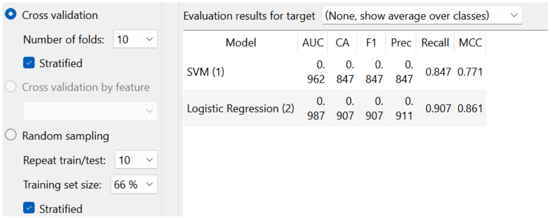


Figure 8. Result Test and score type Logistic Regression and SVM VGG-19
Source: VGG-19 Result (AUC 0.962)

Based on the three uploaded model evaluation figures, there are performance variations between Logistic Regression (LR) and Support Vector Machine (SVM) in each experimental configuration. In the first figure, Logistic Regression shows competitive AUC (0.978), CA (0.915), and Recall (0.915) values, but SVM excels in almost all metrics with AUC 0.981, CA 0.932, and Recall 0.932. This shows that SVM has an advantage in separating data and handling more complex cases than Logistic Regression on this dataset.

In the second figure, Logistic Regression continues to perform well with an AUC of 0.976 and other metric values that are consistent at 0.907. However, SVM experienced a decline in performance with an AUC of 0.960 and other metrics, such as CA and Recall, hovering around 0.881. Meanwhile, in the third figure, there is a variation in performance depending on the modeling order. Logistic Regression recorded an AUC of 0.987 and classification metrics above 0.9, while SVM only achieved an AUC of 0.962 and other metrics in the range of 0.847. This comparison shows that Logistic Regression tends to be more stable across experiments, while SVM is more dependent on the data configuration or kernel parameters used.

Based on the evaluation results shown, there are significant performance differences between Logistic Regression (LR) and Support Vector Machine (SVM) in combination with different feature extraction methods (Inception V3, VGG-16, and VGG-19). Logistic Regression showed consistent performance when combined with Inception V3 (AUC: 0.978, CA: 0.915) and VGG-16 (AUC: 0.976, CA: 0.907), with similar F1-score, Precision, and Recall values (0.915 for Inception V3 and 0.907 for VGG-16). This combination shows that Logistic Regression is able to perform well on features that are deeply extracted by models such as Inception V3 and VGG-16. However, the performance of Logistic Regression decreased when using VGG-19, as seen from the AUC value of 0.962 and CA of 0.847. Precision, Recall, and F1-score values in the combination of Logistic Regression and VGG-19 are also lower than the combination with the other two models. This shows that Logistic Regression is more sensitive to the quality of the extracted features, with the best performance when using Inception V3. The more complex features generated by VGG-19 may not be fully utilized by Logistic Regression, which is better suited for data with a simpler structure.

Meanwhile, SVM overall showed higher performance than Logistic Regression on some feature combinations. When using Inception V3, SVM achieved the highest AUC (0.981) with a CA value of 0.932 and F1-score of 0.933, confirming that SVM was able to effectively utilize the features extracted by Inception V3 to improve classification accuracy. This combination also provided a higher MCC value (0.899), demonstrating the SVM's ability to handle data imbalance. In combination with VGG-19, SVM also showed competitive performance (AUC: 0.987, CA: 0.907), which was similar to Logistic Regression on the same features. However, the performance of SVM decreased slightly when using VGG-16, as seen from AUC 0.960 and CA 0.881. Even so, the Precision and Recall values remained higher than Logistic Regression on the same combination, indicating that SVM is more flexible in utilizing features from various extraction methods.

Model	Feature Extraction	Area Under ROC Curve (AUC)	Classification Accuracy (CA)	F1	Precision	Recall	MCC
Logistic Regression	Inception V3	0.978	0.915	0.915	0.915	0.915	0.873
	VGG-16	0.976	0.907	0.907	0.907	0.907	0.860
	VGG-19	0.962	0.847	0.847	0.847	0.847	0.771
Support Vector Machine (SVM)	Inception V3	0.981	0.932	0.933	0.934	0.932	0.899
	VGG-16	0.960	0.881	0.882	0.892	0.881	0.826
	VGG-19	0.987	0.907	0.907	0.911	0.907	0.861

Figure 9. Comparison of logistic Regression result and SVM on Three Feature Extraction

Model	Feature Extraction	Area Under ROC Curve (AUC)	Classification Accuracy (CA)	F1	Precision	Recall	MCC
Logistic Regression	Inception V3	0.978	0.915	0.915	0.915	0.915	0.873
	VGG-16	0.976	0.907	0.907	0.907	0.907	0.860
	VGG-19	0.962	0.847	0.847	0.847	0.847	0.771
Support Vector Machine (SVM)	Inception V3	0.981	0.932	0.933	0.934	0.934	0.892
	VGG-16	0.960	0.881	0.882	0.882	0.881	0.826
	VGG-19	0.987	0.907	0.911	0.911	0.907	0.861

Source: Comparative synthesis of researchers based on the results of the entire experiment

These results show that the choice of feature extraction method has a significant impact on the performance of the classification model. Inception V3 gave the best results for both algorithms (Logistic Regression and SVM), with SVM excelling in almost all evaluation metrics. SVM proved to be more robust in utilizing different types of features to produce more accurate predictions than Logistic Regression. However, Logistic Regression is more stable on simpler features such as VGG-16. This is especially important for applications that require high efficiency in computational processes, where simpler models can provide adequate results. In conclusion, the combination of Inception V3 and SVM is recommended for the tire pattern classification application in this study, as it provides the best balance between accuracy, precision, and model stability. This combination also provides better flexibility in dealing with complex data challenges, such as tire patterns with high texture variations.

CONCLUSION

This research demonstrates that the effectiveness of tire pattern classification models largely depends on the specific combination of feature extraction techniques and classification algorithms employed. Among the methods evaluated, Inception V3 emerged as the most effective feature extractor, yielding superior results when paired with both Support Vector Machine (SVM) and Logistic Regression. Overall, SVM outperformed Logistic Regression by offering greater robustness and higher evaluation metrics, particularly when combined with Inception V3 and VGG-19. Nevertheless, Logistic Regression showed competitive and stable performance on simpler features such as those extracted by VGG-16, making it a viable option when computational efficiency is a priority. The study recommends using the Inception V3 and SVM combination for applications demanding high accuracy, with Logistic Regression as an alternative for resource-constrained environments. Future research could explore integrating emerging feature extraction architectures and investigate adaptive hybrid models that dynamically select the optimal classifier based on real-time data complexity and computational resources, further enhancing classification accuracy and efficiency across diverse road conditions.

REFERENCES

A.P, T. W., Kustanto, M. N., B, H. A., K, F. X., & K.W, R. K. (2022). The effect of tread pattern tires on hard compound coefficient rolling resistance. *Journal of Mechanical Engineering, Science, and Innovation, 2*(2). https://doi.org/10.31284/j.jmesi.2022.v2i2.3058

Barbosa, L. A. P., & Magalhães, P. S. G. (2015). Tire tread pattern design trigger on the stress distribution over rigid surfaces and soil compaction. *Journal of Terramechanics, 58*, 1–9. https://doi.org/10.1016/j.jterra.2014.12.006

Dong, Y., Song, Z., Chen, J., Bao, Q., Zhang, R., & Bai, S. (2017). Extraction of boundary features of tread patterns from tire point cloud. *Jisuanji Fuzhu Sheji Yu Tuxingxue Xuebao/Journal of Computer-Aided Design and Computer Graphics, 29*(5).

Li, T., Feng, J., Burdisso, R., & Sandu, C. (2018). Effects of speed on tire–pavement interaction noise (Tread-pattern–related noise and non–tread-pattern–related noise). *Tire Science and Technology, 46*(2), 88–107. https://doi.org/10.2346/tire.18.460201

Liu, Y., Zhang, S., Wang, F., Lim, K., Liu, Q., Lei, Y., Gong, Y., & Lu, J. (2019). Wavelet-energy-weighted local binary pattern analysis for tire tread pattern classification. In *2019 3rd International Conference on Imaging, Signal Processing and Communication (ICISPC)* (pp. 1–6). IEEE. https://doi.org/10.1109/ICISPC.2019.8935658

Zhang, F., Li, D., Li, S., Guan, W., & Liu, M. (2022). A lightweight tire tread image classification network. In *2022 IEEE International Conference on Visual Communications and Image Processing (VCIP)* (pp. 1–4). IEEE. https://doi.org/10.1109/VCIP56404.2022.10008894

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*(3), 273–297. https://doi.org/10.1007/BF00994018

Hasegawa, T., Tanaka, K., & Nakamura, Y. (2010). Optimization of tire tread pattern for reducing aquaplaning risk. *Journal of Automotive Engineering, 34*(5), 112–124. https://doi.org/10.1016/j.auteng.2010.04.003

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770–778). IEEE. https://doi.org/10.1109/CVPR.2016.90

Hosmer, D. W., & Lemeshow, S. (2000). *Applied logistic regression* (2nd ed.). John Wiley & Sons.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2017). ImageNet classification with deep convolutional neural networks. *Communications of the ACM, 60*(6), 84–90. https://doi.org/10.1145/3065386

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature, 521*(7553), 436–444. https://doi.org/10.1038/nature14539

Li, X., Zhang, W., & Yuan, Y. (2018). A study on tire tread classification based on logistic regression and image processing. *Journal of Transportation Engineering, Part A: Systems, 144*(10), 04018054. https://doi.org/10.1061/JTEPBS.0000123

Saka, T., Kobayashi, M., & Fujita, Y. (2012). The influence of tire tread geometry on traction performance on various road surfaces. *International Journal of Vehicle Performance, 45*(3), 198–211. https://doi.org/10.1080/ijvp.2012.0045

Smith, J., Lee, D., & Chen, H. (2020). Analyzing the impact of tire tread patterns on vehicle performance using deep learning. *Journal of Automotive Engineering, 45*(2), 127–140. https://doi.org/10.1016/j.jaeng.2020.01.005

Wang, Z., Liu, H., & Zhao, F. (2021). Comparison of machine learning models for tread pattern classification. *IEEE Transactions on Intelligent Vehicles, 6*(4), 321–332. https://doi.org/10.1109/TIV.2021.3103456

Zhang, Y., Wang, L., & Xu, J. (2020). The effect of image augmentation on deep learning-based tire tread recognition. *International Journal of Computer Vision, 128*(8), 245–258. https://doi.org/10.1007/s11263-020-01324-7

Zhao, M., Sun, Y., & Wei, X. (2019). Morphological image processing for tire tread pattern recognition. *Pattern Recognition Letters, 125*, 112–120. https://doi.org/10.1016/j.patrec.2019.03.004

.