



Classification of Environmental and Human Obstacles in Automatic Driving Using EfficientNet

Samaa Khalid Ghanim 

Department of Engineering Electronic and Communication Techniques, Polytechnic College Mosul, Northern Technical University, Mosul 41002, Iraq

Corresponding Author Email: mti.lec22.samaa@ntu.edu.iq

Copyright: ©2026 The author. This article is published by IIETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/jesa.590106>

ABSTRACT

Received: 4 November 2025

Revised: 11 January 2026

Accepted: 20 January 2026

Available online: 31 January 2026

Keywords:

Convolutional Neural Networks, EfficientNetB7, autonomous driving, classification obstacles

Neural networks currently play an important role in the technology of autonomous driving, especially in object classification. The application of deep learning models provides the ability to improve perception, enabling vehicles to become more reliable and capable of differentiating various environmental and human obstacles compared to other approaches. Neural networks and their techniques are essential for improving safety and efficiency in autonomous systems. The classification of obstacles is mainly divided into two types: Environmental and human. The Environmental obstacles include the stationary objects like buildings and the dynamics. The human obstacles are the pedestrians, whose unpredictable motions generate diverse challenges for autonomous systems. Obstacles predicted are detected using trained neural networks on datasets. In this paper, a proposal system has been utilized for the classification of obstacles in autonomous driving using the EfficientNetB7 neural network. The images of the obstacles are collected and then divided into three subsets: training, validation, and testing. The proposed system is tested in terms of performance metrics, showing a precision of 95%, a recall of 95%, and an F1-score of 95%.

1. INTRODUCTION

Detecting and classifying obstacles are vital in the development of autonomous driving systems, as ensuring safety and reliability under various driving conditions is critical. Recent developments and improvements in computer vision and deep learning have led to significantly greater accuracy and robustness in obstacle detection, particularly in complex traffic situations. Early methods of detecting obstacles employed traditional computer vision methods, including background subtraction, edge detection, and engineered features, often combined with traditional classifiers. However, these techniques typically struggled with variations in lighting, occlusions, and reliance on real time performance [1]. With the emergence of deep learning, Convolutional Neural Networks (CNNs) have dominated visual perception tasks [2-3]. Architectures for example YOLO, SSD, and Faster R-CNN is widespread approaches to real-time recognition and classification of objects in self-driving vehicles [1]. These detectors can classify a multitude of road object classes including vehicles, pedestrians, cyclists, traffic signs, and static objects [4-5].

Recently, researchers have examined more sophisticated backbone networks, such as Efficient Net and a Transformer-based architecture DETR, to improve the performance of detection and computational efficiency in embedded systems. Many studies have shown that, when employing a high-performance light-weight model, live processing is possible on edge devices [6-7].

Publicly available datasets such as KITTI, BDD100K, kaggle, and image-CV have facilitated benchmarking of obstacle detection and classification algorithms under diverse conditions. For example, KITTI provides annotated images for detecting cars, pedestrians, and cyclists in urban and rural environments, while BDD100K expands this to include weather variations and nighttime scenes [8].

Some recent works have incorporated multi-modal sensing by combining RGB images with LiDAR point clouds or radar data to enhance the detection and classification of obstacles, especially in adverse weather or low-light conditions [8]. Fusion methods and sensor redundancy show increased reliability and robustness, addressing key challenges for fully autonomous vehicles [9].

Despite these advancements, several challenges remain. Class imbalance, small and rare object detection and domain adaptation for unseen environments continue to be active research areas [10]. Moreover, real-time obstacle classification must balance model complexity with inference speed to meet the stringent latency constraints of autonomous driving systems. To address these gaps, recent research trends include leveraging semi-supervised learning, domain adaptation, and synthetic data augmentation to improve generalization for unseen scenarios [11].

2. RELATED WORK

Mounsey et al. [12] suggested a review of CNN for solving

pedestrian detection and CNN-based models as well as transfer learning models. The VGG-16 architecture-based CNN model is additionally improved using a transfer learning method and proposed a pre-processing scheme used for the preparation of 3D spatial data which are achieved by LiDAR sensors. Such a pre-processing paradigm can classify nominee areas for classification, either using 3D classification or a modified model mixed between 2D and 3D classifications via sensor fusion. A succession of models incorporating transfer learning and CNNs proposed and achieved higher than 98% accuracy with the adaptive transfer learning model.

Islam et al. [13] suggested a deep neural network based autonomous vehicle system, via object detection and semantic segmentation techniques, to eliminate the negative effects of this type of hazard, enabling the autonomous vehicle to navigate safely around these hazards 21% faster than a conventional DNN-based autonomous vehicle.

Huang et al. [14] proposed a Multi-Scale Metric Learning (MSML) model for the classification of obstacles in autonomous vehicles. The MSML model leverages ideas from both metric learning and integrated multi-scale feature fusion. The MSML model can preserve more contextual information from images of obstacles, thereby improving its interpretability as well as the capability of generalization. Metric learning further fine-tunes the model by structuring the feature space such that all samples of the same class are close to each other, and those of a different class are distant from one another. This improves the model's discrimination between various types of obstacles, making it more robust against noise and better generalizable to new data.

The performance of three machine learning algorithms: k-nearest neighbor (kNN), naive Bayesian classifier (NBC), and support vector machine (SVM) is evaluated and tested. The algorithms are trained, and the final performance of the method in a real traffic scenario of 16 pedestrian samples and 469 non-pedestrian samples exhibited sensitivity (81.2%), accuracy (96.2%), and specificity (96.8%) [15].

Gagnaniello et al. [16] suggested a framework to compare multi-object detection and tracking methods on a video of images taken by a camera fixed on a self-driving car. Twenty-two different methods were trained and compared against each other on the BDD100K dataset and their detection,

classification and tracking capacity are tested against a set of evaluation measures. The empirical comparison allowed for the highlighting of the robustness of QDTrack as a tracking method, observing that it is particularly effective with a ConvNext detector, SwinT, or HRNet.

Li et al. [17] presented a novel domain-adaptive object detection paradigm, specifically formulated for intelligent vehicle observation in rain and foggy scenes. The paradigm includes image-level and object-level variations to state the domain shift from global image style to local object appearance. The results determine the effectiveness of the suggested DA approach in enhancing object detection performance and enriching intelligent vehicle perception in challenging rainy and foggy weather conditions.

Chen et al. [18] proposed method of 3D object detection of obstacles in automatic driving which generated a class-specific list of object candidate proposals that go over a typical CNN pipeline to achieve high-grade object detections. In guidance of this aim, energy reduction method is presented in which the object candidates are located in 3D by considering that objects should be on the ground-plane, and afterwards assigns a score to each candidate box using different intuitive potentials that encode semantic segmentation, context information, size and location priors and overall object shape. the investigational assessment demonstrated that the object proposal generation approach significantly beats all monocular methods, and attains the superlative detection performance on the challenging KITTI benchmark, among published monocular competitors.

3. EFFICIENTNET

The EfficientNet was chosen architecture for obstacles in the autonomous car classification model. EffectNet models are a set of CNN that beat various state-of-the-art CNN models such as Inception net, Resnet, Xception, and other [16]. The Efficient Nets suite contains seven models, which are B0 to B7. Efficient Nets models, which are constructed by scaling model difficulty through depth, width, and resolution based on a scaling rule described in Eq. (1). In Eq. (1), α , β , and γ represent constants derived [19].

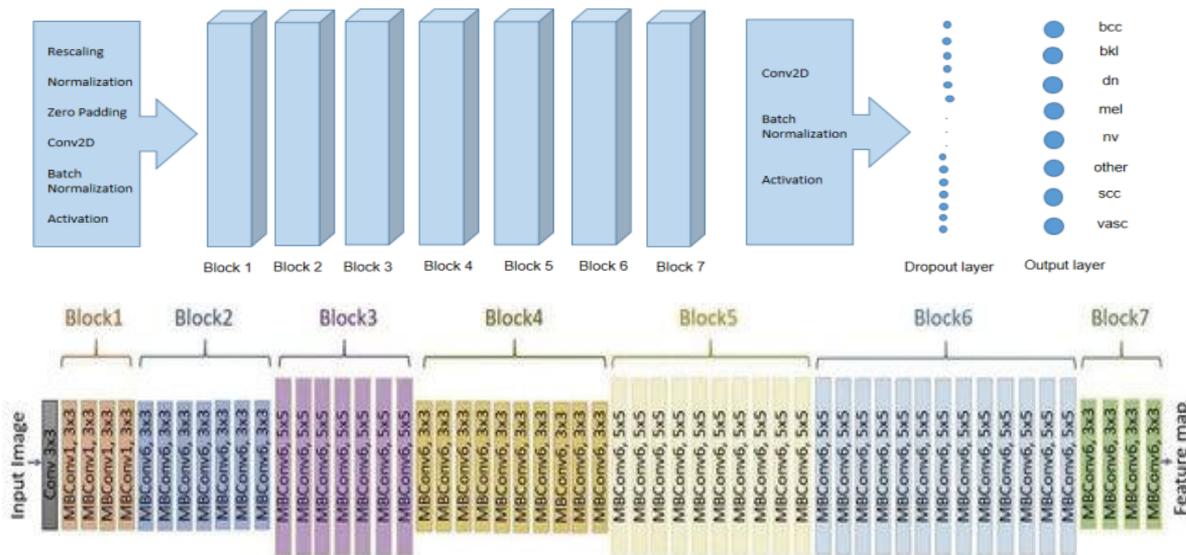


Figure 1. EfficientNet B7 architectural [19, 20]

$$\begin{aligned}
\text{depth} &= \alpha^\phi \\
\text{width} &= \beta^\phi \\
\text{resolution} &= \gamma^\phi \\
\text{s.t. } \alpha^2 \cdot \beta^2 &\approx 2
\end{aligned}
\tag{1}$$

$\alpha \geq 1 \quad \beta \geq 1 \quad \gamma \geq 1$ [15].

While ϕ is the scaling factor.

A simple building block of EfficientNet architecture is mobile inverted bottleneck convolution (MBConv) with squeeze and excitation optimization. EfficientNet network family consists of different number of these MBConv blocks. Increasing number from EfficientNetB0 to EfficientNetB7, depth, width, resolution and model size keeps on increasing and accuracy also gets better. Superlative performer model EfficientNetB7 beats the earlier state-of-art CNNs by accuracy on ImageNet, and is also 8.4× lighter and 6.1× quicker compared to the best CNN now. The EfficientNetB7 network architecture is demonstrated in Figure 1. It is classified into seven blocks depending on filter size, striding, and number of channels [20].

4. METHODOLOGY

The proposed system is implemented to classification obstacles of autonomous driving as shown in Figure 2, The dataset contains obstacles of autonomous driving images consists of 5 groups of images collected from the web-site (images.CV and image google by means of keywords associated to autonomous driving obstacles) and A manual inspection procedure was implemented to eliminate poor quality, repeating, and irrelevant images. Images that were highly blurred, largely occluded, or wrongly labeled were removed. The classes of obstacles consist of Bicycles and Motorbikes, cars, pedestrians in the street, potholes, and traffic lights, as shown in Table 1. The dataset of five class of obstacle splits into percentages of 80% training, 10% validation, and 10% test, Though the value of samples in each class is imbalanced (1028-1357), the imbalance ratio is maintained within the acceptable limits. To overcome the imbalance, data augmentation strategies that are class-aware are employed during the training process as well as the requirement of large amounts of data in CNN to learn useful patterns from images.

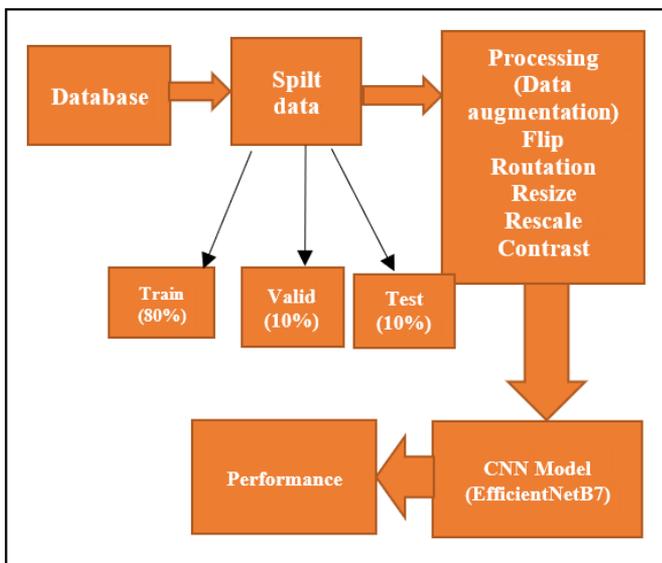


Figure 2. Block diagram of proposed system

Table 1. The statistics of obstacles in automatic driving

Class of Obstacles	Number of Images
Bicycles and Motors	1057
Vehicles	1135
Pedestrians on Street	1028
Potholes	1357
Traffic Light and Poles	1127

The data augmentation was carried out using a sequence of preprocessing operations in TensorFlow/Keras to make the results of the modeling process both robust and generalizable. It is essential to note that an input size of 224×224 pixels is standardized to ensure proper compatibility with most deep learning models. The pixel intensity was normalized by using a scale factor of 1/255 to convert the intensity level [0, 255] to [0, 1], providing stability to the training procedure. To ensure variability of scenes as perceived by human vision, horizontal flipping was carried out to address scenarios arising from object orientation changes. Furthermore, random rotation, zooming, and contrast changes with increments of 0.1, reflecting moderate changes of up to 10% increments, were added to generate diversified versions of the scene while maintaining semantic correctness of images. These operations were carried out on both training and validation datasets while training, effectively controlling overfitting and improving generalization performance of the results of modeling.

The model's images are trained through a pre-trained CNN model is EfficientNetB7. The input layer in the model architecture is inherited directly from the pre-trained model, ensuring the learned features are retained. Data augmentation techniques are utilized on the input images to make the model robust. Based on the pre-trained feature extractor model, a custom classification head is designed for the proposed model architecture. The custom classification head contains two fully connected layers with a neuron count of 128 and 256, respectively. Each connected layer is followed by a batch normalization technique to make the model converge faster. Dropout with a value of 0.45 is used to avoid overfitting in the model. The last output layer has a softmax activation function with five units, reflecting the number of classes in the targets for multi-class probability estimation. The model was optimized with the Adam optimizer at a learning rate of 1×10^{-5} for fine-tuning in transfer learning, while the loss function was set to categorical cross-entropy for multi-class classification tasks. Evaluation of the model was done using classification accuracy as the criteria. Training was done for a maximum of 100 epochs on both the training set and validation set with a host of callbacks such as early stopping to prevent overfitting, reducing the learning rate for better convergence, a model saves for the best-performing models with the finest weights, as well as TensorBoard logging to track the training. the Early Stopping callback was set to monitor the validation loss (val_loss) with a patience of epochs [5], which meant that training would be stopped if no improvement was seen over [5] epochs. The model weights associated with the best performance on the validation set were then loaded at the end of the training process. Finally, a batch size of 32 is a compromise between the stability of the train process and the limitations of the GPU memory. A batch size of 32 ensured a stable convergence point without triggering a memory overflow problem, especially for the high input resolution of the EfficientNetB7 model. Table 2 explains the setting of proposal system.

Table 2. The setting of proposal system

No. of Epoches	100
Batch Size	32
Learning Rate	1×10^{-5}
Patience Value	5
Run (Acculator)	GUP-100 in kaggle

5. RESULTS AND PERFORMANCE METRICS

The proposal model is tested to classify five classes of different types of obstacles for autonomous driving classification model. Figure 3 displays the classification accuracy of the suggested model for the validation set and training set over 50 epochs. It can be seen that the model acquires rapid improvement rate in accuracy in the epochs followed by an extremely slow stabilization process. The validation accuracy becomes stabilized at around 93.12% during roughly around 40 epochs and then remains almost at the same level, which indicates excellent generalization performance. Meanwhile, the training accuracy continues to increase steadily and converges close to the validation accuracy towards the end of the training process. Whereas Figure 4 shows the loss value of validation and training data where the training loss value is approximately 0.2521 but the validation loss value is 0.2455 at epoch 50. The results overall prove that the model is well-trained with very little overfitting, as the training and validation accuracy and loss curves are quite close to one another and both show good levels of performance. This is an affirmation of the success of the model design and training method using EfficientNetB7 in the classification task. Also, predictions are made on the test data and the results were obtained where test Loss is 0.18240 and test Accuracy is 0.9526.

Some of the metrics are employed in the performance evaluation of autonomous driving classification models. Three binary performance metrics were employed: precision, Recall and F1-score to evaluate the performance of the model of individual obstacles for autonomous driving condition as follows:

Precision (P): determine the accuracy of positive predictions. Can be calculated using Eq. (2).

$$P = \frac{T_p}{T_p + F_p} \tag{2}$$

Recall (R):

The percentage of all actual positive results that were correctly classified as positive. Can be calculated from Eq. (3).

$$R = \frac{T_p}{T_p + F_N} \tag{3}$$

F1-score (F1):

The harmonic mean between recall and precision. It takes the average of the two measures together, which is useful if recall and precision are in conflict. Can be calculated from Eq. (4).

$$F1 = 2 * \frac{T_p * F_N}{T_p + F_N} \tag{4}$$

where, T_p : true positives, F_p : false positives, F_N : false negatives.

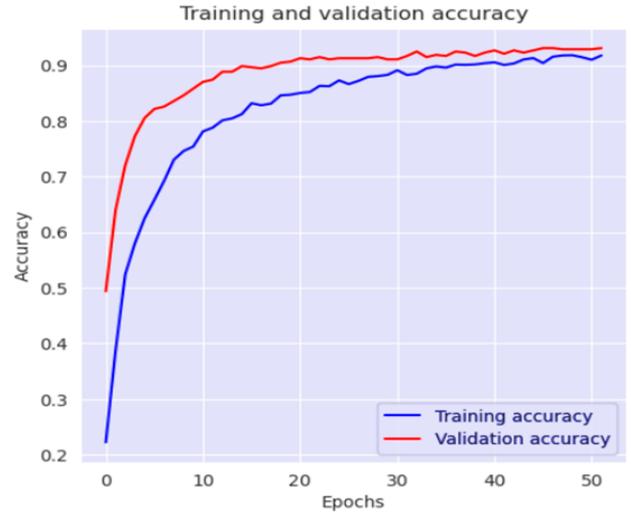


Figure 3. The curve of training and validation accuracy

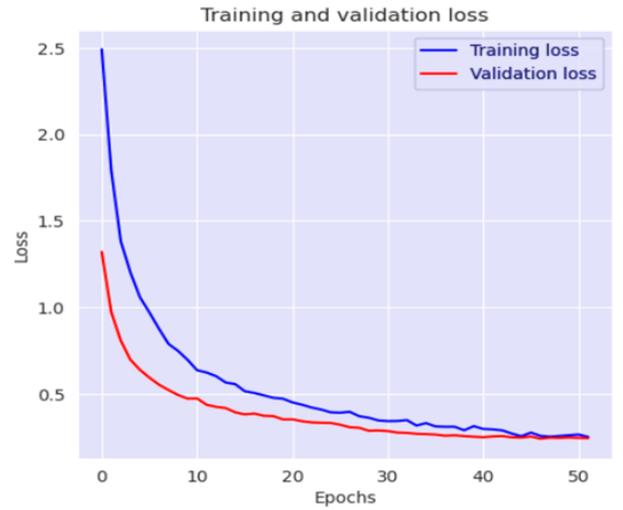


Figure 4. The curve of training and validation loss

	precision	recall	f1-score
Bicycle and Motor	0.960396	0.970000	0.965174
Pedestrians on the street	0.904762	0.926829	0.915663
Potholes	0.978723	0.978723	0.978723
Traffic lights and poles	0.967742	0.918367	0.942408
vehicles	0.938462	0.953125	0.945736

Figure 5. Classification report of obstacles

As shown in Figures 5 and 6, the classification report of performance metrics and confusion matrix are two important tools used for quantifying the performance of an image classification model.

Whereas A classification report is a summary of the most important performance metrics for a classification model, containing precision, recall, and F1-score, and the model accuracy. It can be readily seen that the model attains F1-score of 90 and above for most of the classes. In which the highest 97% accuracy was realized by the Potholes group, and the

Pedestrians on the street group realized the lowest 90% accuracy.

Confusion matrix is a table that is employed to tally the number of correct and incorrect predictions provided by a classification model over a set of test data. The entries of the matrix are indicative of the number of the test samples belonging to a specific class, and how many of them were predicted otherwise by the model.

The diagonal elements are the properly predicted samples. In Bicycles and Motors, a total of 97 samples were correctly predicted out of the total 100 samples as shown in Figure 6 that occur between visually and contextually similar classes, such as Pedestrians on Street, which share background characteristics. In Pedestrians on Street, A total of 76 samples were correctly predicted out of the total 82 samples. There are similarities with other classes such as Bicycles and Motors class (2 samples), vehicles class (2 samples), Potholes class (1 sample), and traffic light and Poles class (1 sample). It can be used to identify specific spots where the model is making mistakes, and to fix problems with model predictions to provide a detailed analysis of the performance of the model, such as accuracy, precision, recall, and F1-score for each class.

The images of different categories of obstacles were tested for the obstacle classification of the autonomous driving for

the verification of the accuracy of the classification of the proposed system, as shown in Figure 7. The obstacle images classified into different categories mostly agreed on the prediction.

The performance evaluation of the EfficientNetB7 model demonstrates a clear between accuracy and computational efficiency overall with precision = 95%, recall = 95%, and F1-score = 95%. As well as the model achieves an inference time of 0.0089seconds per image, corresponding to a throughput of 112.38 FPS, while requiring 10.44 GFLOPs. These results indicate that EfficientNetB7 delivers strong computational performance despite its deep and complex architecture. The relatively moderate inference time suggests that the model is suitable for near real-time applications on high-performance hardware, while the achieved FPS reflects efficient utilization of computational resources. Although the FLOPs value is higher compared to lightweight architectures, it remains acceptable for applications where accuracy is prioritized over strict real-time constraints. Overall, the results confirm that EfficientNetB7 provides a balanced compromise between high representational capacity and computational cost, making it a viable choice for safety-critical and vision-based intelligent systems as shown in Table 3.

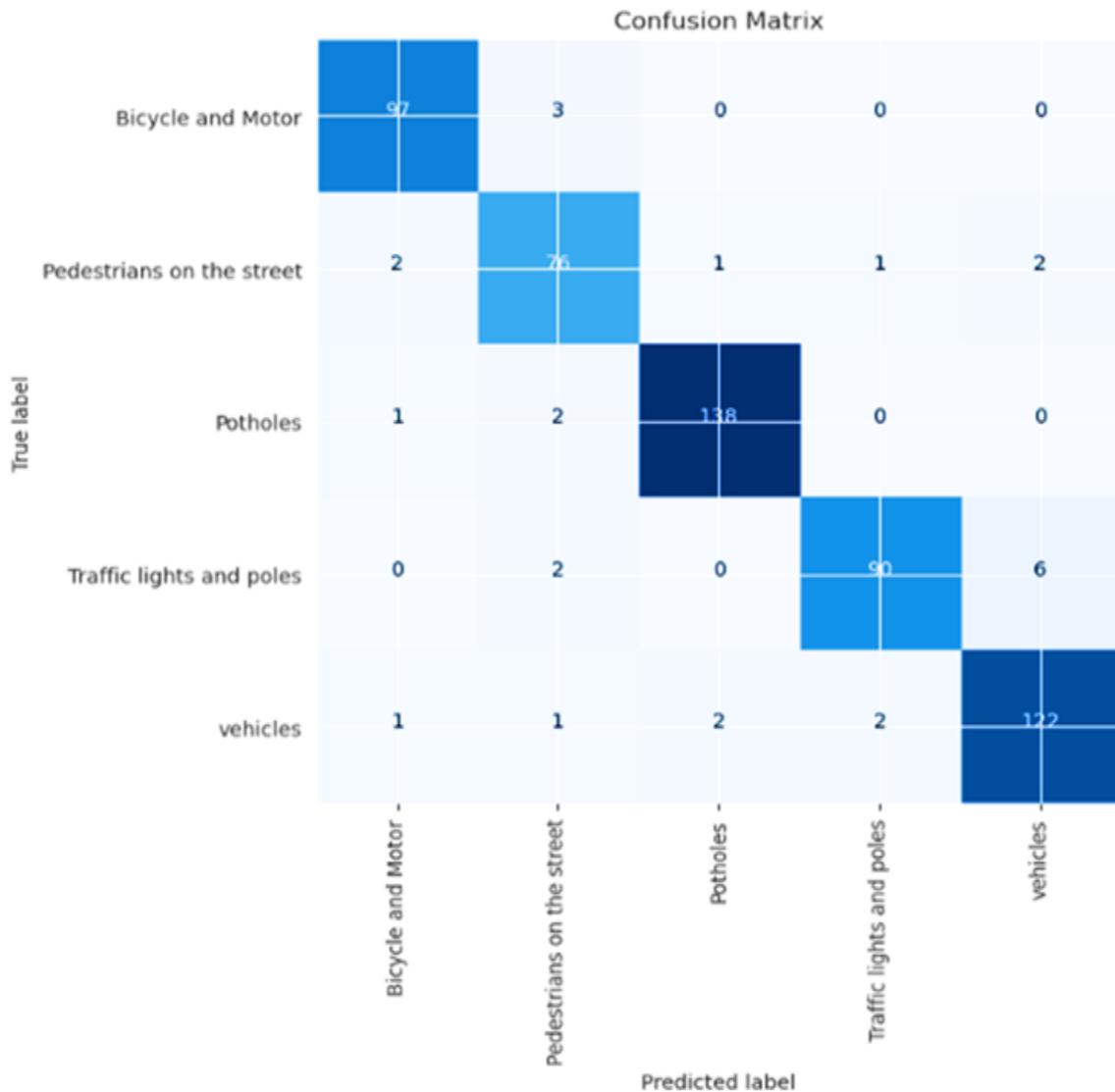


Figure 6. Confusion matrix



Figure 7. Testing the predicted with true of images

Table 3. The performance of proposal system using EfficientNetB7

Model	EfficientNetB7
Params	64461340
FLOPs (G)	10.44
Inference Time (s) per image	0.0089
FPS	112.38
precision	95%
recall	95%
F1-score	95%

Precision, and Recall metrics, with no statistically significant differences observed.

Table 4. The performance of proposal system using EfficientNetV2

Model	EfficientNetV2
FLOPs (G)	3.23
Inference Time (s) per image	0.0038
FPS	262.63
precision	95%
recall	95%

In addition to this, other experiments were done using a different model: efficientNetV2, with results as indicated in Table 4.

The results indicate that EfficientNetB7 and EfficientNetV2 achieved highly comparable performance across all evaluated

But comparing the performance analysis of both EfficientNetV2 and EfficientNetB7 with metrics inference time, FPS, model parameters, and computational complexity (FLOPs), there is a considerable improvement in efficiency in

the use of EfficientNetV2. In this case, there is a considerable difference in inference time, throughput, and computational cost as indicated by 0.0038 seconds per image inference time and a throughput of 262.63 FPS with a computational cost of 3.23 GFLOPs by EfficientNetV2. On the other hand, there is a higher inference time of 0.0089 seconds and a lower throughput of 112.38 FPS along with a considerably large computational cost of 10.44 GFLOPs by EfficientNetB7. This indicates the efficiency of EfficientNetV2 is over two times faster compared to EfficientNetB7 and a decrease in computational complexity of nearly three times. The considerable improvement in inference time and a reduction in FLOPs indicate efficiency and effectiveness of the model in real-time and resource-constrained systems. At the same time, EfficientNetB7 can prove to be an optimal choice for tasks where the highest representational accuracy is of utmost priority rather than computational competency.

6. CONCLUSION

Obstacle image classification in automatic driving is a complex task that involves using artificial intelligence (AI) algorithms to identify and categorize obstacles based on their visual characteristics. The advancement in deep learning and computer vision technologies, it is now possible to develop highly accurate and efficient obstacle image classification systems that can analyze millions of images in real-time. The proposal system proves the excellent performance using EfficientNetB7 neural network in precision = 95% and according to the validation data accuracy values and loss of the validation data in curves which shows a valid accuracy of 0.9312 and valid loss of 0.2455 at 50th epoch. Furthermore, the test accuracy was 0.9524 and test loss was 0.1826 of testing data. The confusion matrix and classification report also shows through has precision of 90 and above for all the classes, which the Potholes class achieving a maximum accuracy of 97% and the Pedestrians on the street class achieving a minimum accuracy of 90%, classification model showed a good performance of image.

REFERENCES

- [1] Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M. (2020). Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv: 2004.10934. <https://doi.org/10.48550/arXiv.2004.10934>
- [2] Ahmed, H.A., Mohammed, E.A. (2022). Using Artificial Intelligence to classify osteoarthritis in the knee joint. NTU Journal of Engineering and Technology, 1(3): 31-40. <https://doi.org/10.56286/ntujet.v1i3.155>
- [3] Shah, V., Sajjani, N. (2020). Multi-class image classification using CNN and tflite. International Journal of Research in Engineering, Science and Management, 3(11): 65-68. <https://doi.org/10.47607/ijresm.2020.375>
- [4] Ren, S., He, K., Girshick, R., Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. Advances in Neural Information Processing Systems, 28. <https://doi.org/10.48550/arXiv.1506.01497>
- [5] Trivedi, J., Devi, M.S., Dhara, D. (2021). Vehicle classification using the convolution neural network approach. Zeszyty Naukowe. Transport/Politechnika Śląska. <https://doi.org/10.20858/sjsutst.2021.112.7.16>
- [6] Tan, M., Le, Q. (2019). EfficientNet: Rethinking model scaling for convolutional neural networks. In International Conference on Machine Learning, pp. 6105-6114. <https://doi.org/10.48550/arXiv.1905.11946>
- [7] Ahdi, M.W., Kunaefi, A., Nugroho, B.A., Yusuf, A. (2023). Convolutional Neural Network (CNN) EfficientNet-B0 model architecture for paddy diseases classification. In 2023 14th International Conference on Information & Communication Technology and System (ICTS), Surabaya, Indonesia, pp. 105-110. <https://doi.org/10.1109/ICTS58770.2023.10330828>
- [8] Ku, J., Mozifian, M., Lee, J., Harakeh, A., Waslander, S.L. (2018). Joint 3D proposal generation and object detection from view aggregation. In 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Madrid, Spain, pp. 1-8. <https://doi.org/10.1109/IROS.2018.8594049>
- [9] Chen, X., Ma, H., Wan, J., Li, B., Xia, T. (2017). Multi-view 3D object detection network for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 1907-1915. <https://doi.org/10.48550/arXiv.1611.07759>
- [10] Saito, K., Watanabe, K., Ushiku, Y., Harada, T. (2018). Maximum classifier discrepancy for unsupervised domain adaptation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 3723-3732. <https://doi.org/10.48550/arXiv.1712.02560>
- [11] Caesar, H., Bankiti, V., Lang, A.H., Vora, S., Liong, V.E., Xu, Q., Krishnan, A., Pan, Y., Baldan, G., Beijbom, O. (2020). nuScenes: A multimodal dataset for autonomous driving. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 11621-11631. <https://doi.org/10.48550/arXiv.1903.11027>
- [12] Mounsey, A., Khan, A., Sharma, S. (2021). Deep and transfer learning approaches for pedestrian identification and classification in autonomous vehicles. Electronics, 10(24): 3159. <https://doi.org/10.3390/electronics10243159>
- [13] Islam, M., Chowdhury, M., Li, H., Hu, H. (2019). Vision-based navigation of autonomous vehicles in roadway environments with unexpected hazards. Transportation Research Record, 2673(12): 494-507. <https://doi.org/10.1177/0361198119855606>
- [14] Huang, X., Liu, J., Zeng, Q. (2024). A multi-scale metric learning model for obstacle classification in autonomous driving. IEEE Internet of Things Journal, 12(8): 9848-9857. <https://doi.org/10.1109/JIOT.2024.3509395>
- [15] Navarro, P.J., Fernandez, C., Borraz, R., Alonso, D. (2016). A machine learning approach to pedestrian detection for autonomous vehicles using high-definition 3D range data. Sensors, 17(1): 18. <https://doi.org/10.3390/s17010018>
- [16] Gragnaniello, D., Greco, A., Saggese, A., Vento, M., Vicinanza, A. (2023). Benchmarking 2D multi-object detection and tracking algorithms in autonomous vehicle driving scenarios. Sensors, 23(8): 4024. <https://doi.org/10.3390/s23084024>
- [17] Li, J., Xu, R., Liu, X., Ma, J., Li, B., Zou, Q., Ma, J., Yu, H. (2024). Domain adaptation based object detection for autonomous driving in foggy and rainy weather. IEEE Transactions on Intelligent Vehicles, 10(2): 900-911.

- <https://doi.org/10.1109/TIV.2024.3419689>
- [18] Chen, X., Kundu, K., Zhang, Z., Ma, H., Fidler, S., Urtasun, R. (2016). Monocular 3D object detection for autonomous driving. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2147-2156. <https://doi.org/10.48550/arXiv.2203.02112>
- [19] Mhedbi, R., Chan, H.O., Credico, P., Joshi, R., Wong, J.N., Hong, C. (2023). A Convolutional Neural Network based system for classifying malignant and benign skin lesions using mobile-device images. medRxiv. <https://doi.org/10.1101/2023.12.06.23299413>
- [20] Baheti, B., Innani, S., Gajre, S., Talbar, S. (2020). Eff-UNet: A novel architecture for semantic segmentation in unstructured environment. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, Seattle, WA, USA, pp. 358-359. <https://doi.org/10.1109/CVPRW50498.2020.00187>