Critical Vehicle Detection for Intelligent Transportation Systems

Erkut Akdag[*], Egor Bondarev and Peter H. N. De With
Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven 5612AZ, The Netherlands

Keywords: Vehicle Detection, Vehicle Classification, Vision in Transformer, CNN, Intelligent Transportation.

Abstract: An intelligent transportation system (ITS) is one of the core elements of smart cities, enhancing public safety and relieving traffic congestion. Detection and classification of critical vehicles, such as police cars and ambulances, passing through roadways form crucial use cases for ITS. This paper proposes a solution for detecting and classifying safety-critical vehicles on urban roadways using deep learning models. At present, a large-scale dataset for critical vehicles is not publicly available. The appearance scarcity of emergency vehicles and different coloring standards in various countries are significant challenges. To cope with the mentioned drawbacks and to address the unique requirements of our smart city project, we first generate a large-scale critical vehicle dataset, combining images retrieved from various sources with the support of the YOLO vehicle detection model. The classes of the generated dataset are: fire truck, police car, ambulance, military police car, dangerous truck, and standard vehicle. Second, we compare the performance of the Vision in Transformer (ViT) network against the traditional convolutional neural networks (CNNs) for the task of critical vehicle classification. Experimental results on our dataset reveal that the ViT-based solution reaches an average accuracy and recall of 99.39% and 99.34%, respectively.

1 INTRODUCTION

Surveillance cameras are increasingly adopted to observe traffic flow in public places to improve public safety and reduce traffic congestion. In the event of high travel demand, or a dangerous traffic incident, traffic congestion is likely to happen, especially in urban places. For instance, at large intersections or in cities with a high traffic density, accidents arise frequently. In case of an accident, a rapid medical transfer of the affected people may be necessary, where even a short delay can be lethal. Considerable delay in the first response emergency services can occur due to various reasons. For example, emergency vehicles, such as ambulances, fire trucks, and police cars, get occasionally trapped in traffic jams, thereby increasing overall response time. In addition to the accidents, trucks with dangerous cargo in city centers threaten traffic flow and public safety. To address these problems, intelligent transportation systems (ITS) propose beneficial solutions via automated traffic regulation and notification systems. Detection and tracking of critical vehicles (emergency vehicles and trucks with dangerous loads) become an essential part of such a solution.

Artificial neural networks (ANN), support vector machines (SVM), and deep learning studies mainly concentrate on regular vehicle classes, such as bus, auto, cyclist, and van. At present, there is limited research work on critical vehicle detection. The existing studies focus on subsets of emergency vehicles, i.e. excluding trucks carrying dangerous goods or military police cars. However, inclusion of these classes in the critical vehicle landscape is crucial. Critical vehicles in this paper are a combination of emergency vehicles (ambulances, fire trucks, police cars, and military police cars) and trucks with dangerous goods, named "dangerous trucks" throughout the paper.

This study aims at developing models for the critical vehicle detection problem, based on state-of-the-art networks (e.g., YOLOv5, ViT, EfficientNet, and ResNet) as a first step. To this end, we construct a large-scale dataset by the YOLOv5 detection model and label correction. As a second step, we generate critical vehicle models by applying three different classification architectures: EfficientNet, ResNet-50, and ViT, and evaluate the experimental results. All models are publicly available (Wightman, 2019; Glenn Jocher, 2021).

This paper is organized as follows. Section 2 provides an overview of the related work, including vehi-

[*]Corresponding Author

Akdag, E., Bondarev, E. and N. De With, P.
Critical Vehicle Detection for Intelligent Transportation Systems.
DOI: 10.5220/0010968900003191
In *Proceedings of the 8th International Conference on Vehicle Technology and Intelligent Transport Systems (VEHITS 2022)*, pages 165-171
ISBN: 978-989-758-573-9; ISSN: 2184-495X

cle detection and classification for different use cases. Section 3 explains the dataset creation and model generation approaches. The experimental results are evaluated in Section 4, while section 5 concludes the paper.

2 RELATED WORK

This section presents a literature overview on the vehicle detection and classification problem. First, the existing deep learning models are introduced for the object detection task. Second, the studies targeting the detection and classification of regular vehicles are discussed. Last, an overview is provided of existing studies on the detection and classification of emergency vehicles.

2.1 Deep Learning Methods for Object Detection

Deep neural network solutions are widely exploited in visual recognition tasks such as detection and classification. The power of deep neural networks comes from the hierarchical representation of the features, which can be developed during the model training phase. CNNs are able to learn and extract rich and meaningful features that can be utilized for various visual recognition tasks, such as classification, image segmentation, and object detection.

In the early studies on object detection, the region-based convolutional neural network (RCNN) model (Girshick et al., 2014) was proposed. The performance of the RCNN object detector can be improved significantly by applying heavier and deeper CNN models at the feature extraction stage, such as the VGG network (Simonyan and Zisserman, 2014) and the residual network called ResNet-50 (He et al., 2016), containing 50-layers. After the emergence of the YOLO architecture, urban traffic solutions associated with object detection have switched to YOLO models, since the inference time of the YOLOv3 model is faster than the RCNN model. The work in (Redmon and Farhadi, 2018) concludes that YOLOv3 is one of the fastest models for object detection, where it can process 45 images per second. Therefore, in real-time applications, YOLO or the single-shot detector (SSD) (Liu et al., 2016) approaches are predominantly considered. In the literature, YOLOv3 (Redmon and Farhadi, 2018) is extensively practiced for object detection in surveillance, including the vehicle detection problem.

2.2 Detection of Regular Vehicle Types

The detection and classification of regular vehicles, such as cars, buses, and trucks, are mainly studied by performing experiments on different regular vehicle datasets.

In the study by (Arinaldi et al., 2018), the authors implement two different methods for the detection and classification of regular vehicles. The first method is a combination of Mixture of Gaussian (MoG) (Reynolds, 2009) and support vector machine (SVM) (Suykens and Vandewalle, 1999), while the second one is FasterRCNN (Ren et al., 2015). FasterRCNN outperforms MoG in vehicle detection, and it further surpasses SVM for the task of classifying the vehicle types. Authors report results on the Indonesian Toll Road dataset (Arinaldi et al., 2018), and the public MIT traffic dataset (Wang et al., 2008), designed to perform traffic scene analysis and study crowded scenes. In (Suhao et al., 2018), authors concentrate on regular vehicle classes such as car, minibus, and SUV, not covering any critical vehicles. The FasterRCNN (Ren et al., 2015) architecture is applied to perform the vehicle detection and classification task. The VGG16 model is practiced to detect vehicle types in various traffic scenarios. The MIT (Dollar et al., 2011) and Caltech car datasets (Kobayashi et al., 2007) are utilized for model generation. In the work by (Shekade et al., 2020), the authors deploy a dataset with regular vehicles, which involves eleven types of vehicles including bicycle, bus, car, truck, and motorbike. In that research, the YOLOv3 approach is proposed for the vehicle detection task. In the experiments, the total variety of vehicle motion at a specific time is examined with the support of counting and classifying vehicles. Authors conduct their experiments on the MIO vision traffic dataset (Jung et al., 2017).

2.3 Detection of Emergency Vehicle Types

The task of emergency vehicle detection and classification imposes two specific challenges. Emergency vehicle datasets are scarce compared to the regular vehicle datasets. It is challenging to gather data for the emergency vehicle classes due to their rare occurrence on the public roads. Another challenge is that the visual appearances of emergency vehicles vary between countries, for instance, the stripes or colors can be different in each country.

One of the studies focusing on emergency vehicles is performed by (Roy and Rahman, 2019). The authors propose a system to detect emergency vehi-

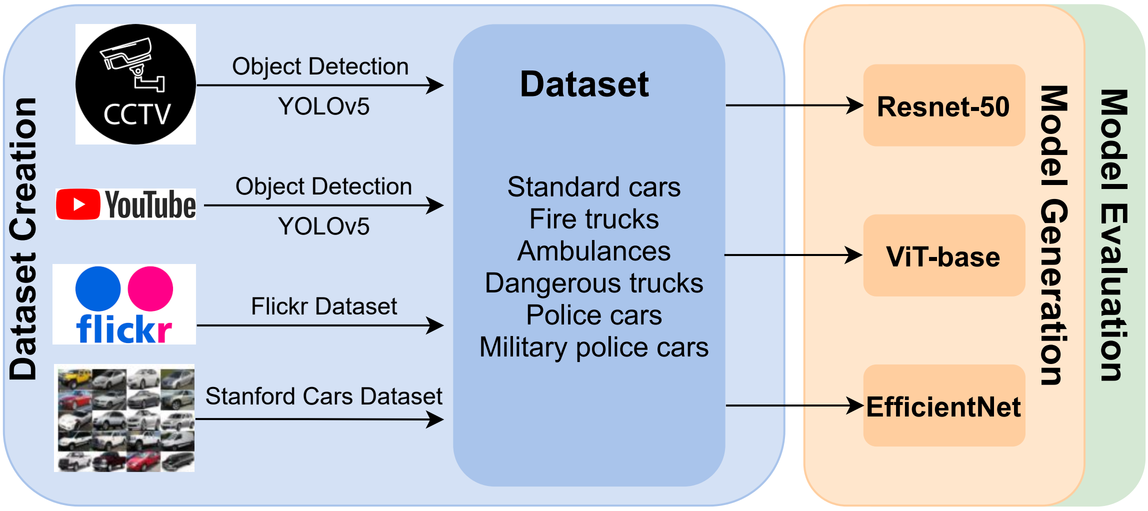

Figure 1: Proposed approach for critical vehicle detection, consisting of the vehicle sample generation from CCTV cameras, YouTube videos, concatenation with Flickr and Stanford Cars dataset, followed by model generation and evaluation with ResNet-50, ViT-base and EfficientNet models.

cles in traffic congestion, which helps the control system to handle traffic flow and prioritize emergency vehicles. In this study, vehicles are first detected from CCTV footage using a deep convolutional network, and they are classified afterwards. YOLOv3 is selected for the vehicle detection task to categorize truck, bus, and car classes, as provided in the COCO dataset (Lin et al., 2014). These classes are passed through the VGG-16 classifier to identify the vehicle type to be emergency or not. Authors conduct their experiments on the Stanford University cars dataset (Szegedy et al., 2015) for regular vehicles and the images collected from the internet for emergency vehicles. Another study by (Carvalho Barbosa et al., 2020) concentrates on the detection and classification of a few emergency vehicles, such as ambulances, fire trucks, and police cars, not including military police cars and dangerous trucks. The authors propose a model named priority vehicle image detection network (PVIDNet) based on YOLOv3 by applying a soft-root-sign (SRS) activation function to decrease the execution time of the proposed model. A database of Brazilian vehicle images is deployed (Carvalho Barbosa et al., 2020) for testing.

As can be derived, previous studies mainly focus on the detection and classification of regular vehicles, and only a few works focus on emergency vehicles. In existing research on emergency vehicles, datasets are limited, and multi-label classification is rarely studied. We aim to generate a large-scale critical vehicle dataset to advance further in this problem domain. A detailed explanation of our dataset is given in Section 3.

3 APPROACH

Figure 1 illustrates the proposed critical vehicle detection approach in two steps, namely the dataset creation and model generation.

3.1 Dataset Creation

For model development and testing, the generated large-scale dataset includes several open-source datasets and images retrieved from CCTV recordings.

First, we adopt the YOLOv5 model instead of YOLOv3 used in previous studies to enhance reliable and robust object detection. Significant differences between YOLOv5 and its prior releases are mosaic data augmentation and auto-learning bounding-box anchors. Likewise, YOLOv5 is nearly 90% lighter than YOLOv4 (Bochkovskiy et al., 2020) in terms of computational cost, while maintaining the detection accuracy and still providing the competitive results. Apart from that, YOLOv5 is one of the fastest deep learning models according to EfficientDet (Tan et al., 2020). Experiments on a GTX 1080 GPU show that YOLOv5 achieves an inference time average of 106 frames/second.

Our dataset includes emergency vehicles and dangerous trucks having a rounded cylindrical shape and transporting dangerous goods, such as chemicals and gasoline. A subset of the Flickr dataset makes a significant contribution to all classes of the created dataset. Moreover, we have searched among various YouTube traffic videos for obtaining Dutch emergency vehicle classes, such as ambulance, police, and

Table 1: Sample volumes for each class in the dataset generated from the Flickr, Stanford cars, YouTube, and CCTV sources.

	Data sources				Generated data		
Class name	Flickr	Stanford Cars	YouTube	CCTV	Total	Train	Test
Standard cars	1,343	5,000	-	102	6,445	5,156	1,289
Fire trucks	1,087	-	-	-	1,087	870	217
Ambulances	200	-	166	70	431	345	86
Police cars	621	-	85	30	736	589	147
Military police cars	363	-	117	-	480	384	96
Dangerous trucks	1,245	-	-	-	1,245	996	249

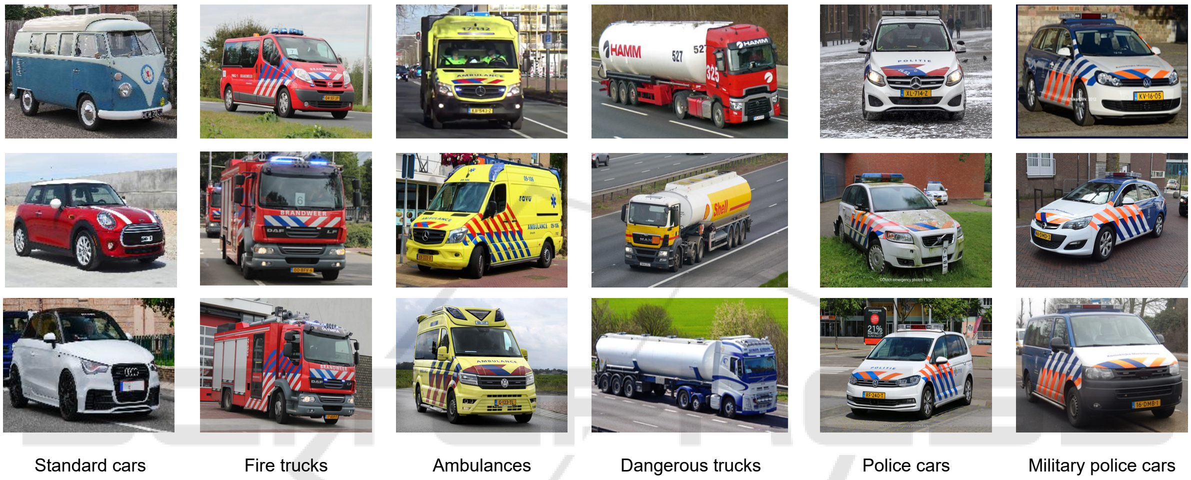

Figure 2: Samples of different vehicle classes in the generated dataset from left to right: standard cars, fire trucks, ambulances, dangerous trucks, police cars, and military police cars.

military police. We apply the YOLOv5 model on the selected videos to increase the number of samples for the specified emergency classes in our generated dataset. To extend the scale of the standard car class, we have combined the publicly available Stanford University cars dataset with our dataset. Besides, we have carefully examined open-source CCTV cameras at different locations of the Netherlands to retrieve as many critical vehicle images as possible. The above-mentioned CCTV video recordings provide further samples of standard cars, ambulances, and police cars through applying the YOLOv5 model for extraction and selection of car instances. The dataset is completed by applying vehicle filtering and annotation processes on the generated image samples for all classes.

Finally, the generated dataset includes 10,424 images classified into 6 categories: standard cars, fire trucks, ambulances, police cars, military police cars, and dangerous trucks. The sample volumes of each class are detailed and listed in Table 1. Additionally, various samples of different vehicle classes are illustrated in Figure 2. We have split the generated dataset into 80% for training and 20% for testing purposes. Furthermore, standard data augmentation techniques are applied to enlarge the available dataset and improve our model.

3.2 Model Generation

Having the large-scale dataset of critical vehicles available, we experiment with three architectures (ViT, EfficientNet, and ResNet) to design an accurate multi-class critical vehicle detection model. The transformer architecture has become the de-facto standard for natural language processing applications. It outperforms other architectures, such as LSTMs (Greff et al., 2016) and gated recurrent units (GRUs) (Dey and Salem, 2017). As opposed to CNNs, a transformer can be applied to the sequence of image patches in the image classification tasks. In the ViT model, images are split into patches like tokens in an NLP application, and sequences of linear embeddings provide the input to the transformer

Table 2: Comparison of image classification performance metrics recall, accuracy, F_1 score for six vehicle classes on the EfficientNet, ViT-base, and ResNet-50 models and the metric averages.

	EfficientNet			ViT-base			ResNet-50		
Class name	Recall	Accuracy	F_1 Score	Recall	Accuracy	F_1 Score	Recall	Accuracy	F_1 Score
Standard cars	**100.00**	**99.92**	**99.96**	99.92	99.84	99.88	99.76	**99.92**	99.84
Fire trucks	**100.00**	99.54	99.77	**100.00**	**100.00**	**100.00**	**100.00**	99.54	99.77
Ambulances	98.83	**100.00**	99.41	**100.00**	**100.00**	**100.00**	98.83	96.59	97.70
Police cars	95.91	97.24	96.57	**98.63**	**98.63**	**98.63**	**98.63**	96.66	97.64
Military police cars	94.79	94.79	94.79	**97.91**	**97.91**	**97.91**	94.79	97.84	96.29
Dangerous trucks	**99.59**	99.20	99.39	**99.59**	**100.00**	**99.79**	**99.59**	**100.00**	**99.79**
Average	98.18	98.44	98.31	**99.34**	**99.39**	**99.35**	98.60	98.27	98.36

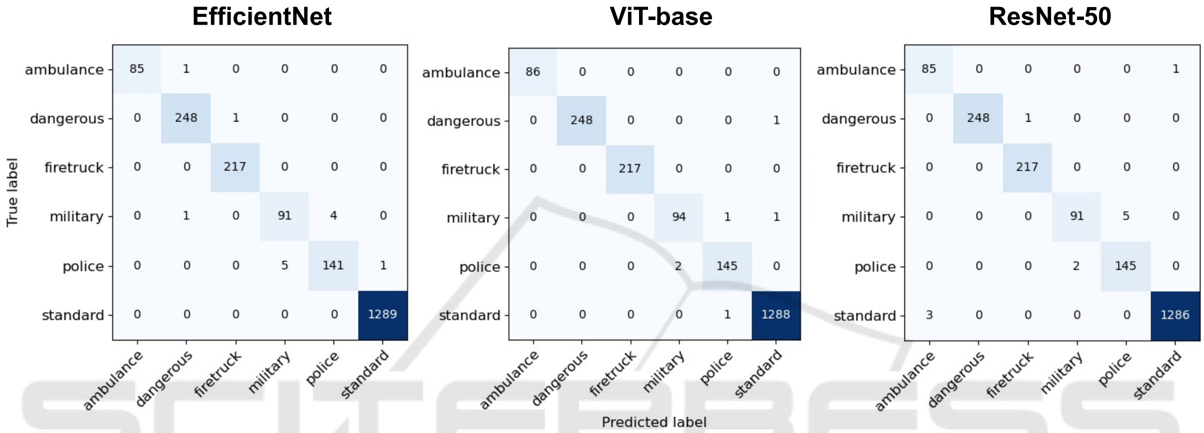

Figure 3: Confusion matrices combining the ground truth and predicted labels for each class using the EfficientNet, ViT-base and ResNet-50 models.

encoder. Positional information is also retained by adding the position embeddings, which are learned without hard-coded vector positions. Lastly, the sequences of the patch and positional embedding vectors are supplied to the input stage of the transformer encoder. There is also a special token at the start of the ViT model, similar to bidirectional encoder representations from transformers (BERT) (Devlin et al., 2018). We practice the VITbase-21k model trained on Imagenet-21k. Furthermore, we experiment with the EfficientNet and ResNet-50 models, which are two state-of-the-art CNN models commonly used for the classification.

We train the models by hyperparameter tuning, where throughout our experiments, we have used an initial learning rate of 0.0002, reduced by one-tenth at every 3 epochs. All models have been trained for a total of 12 epochs using the stochastic gradient descent (SGD) algorithm and the PyTorch framework (Paszke et al., 2019).

4 RESULTS AND EVALUATION

To evaluate the performance of the three models, we deploy the following common metrics: recall, accuracy, and F_1 score. The F_1 score is an appropriate metric for a class-imbalanced dataset similar to ours, since it is the harmonic mean of the recall and precision. It presents the balance between how many examples of actual vehicles are classified correctly and how likely the classification is correct. A commonly used strategy for performing experimental tests is to combine pre-training with fine-tuning to better leverage from the limited amount of vehicle images and improve the classification performance.

The results are summarized in Table 2. In this table, we present six rows for each of the vehicle classes. Three primary columns (EfficientNet, ViT-base, and ResNet-50) show the performance of each corresponding classification model, indicating recall, accuracy, and F_1 score metrics. From the experimental results shown in Table 2, the average accuracy of the ViT model can reach 99.39%, while EfficientNet and ResNet-50 reach 98.44% and 98.27%, respectively. Although the CNN models have better results in a few class types, the ViT model achieves the best experimental results for all metrics on average. Experimental results on the ambulance and fire truck classes are outstanding, since their color and stripes are salient and distinguishable. Likewise, the metrics on the standard car class are high, since regular cars do not have specific striping or coloring patterns. However, these results are slightly lower for military police and police cars. In terms of the F_1 score metric, military police cars achieve the lowest total score with 94.79% in EfficientNet and 96.29% in ResNet-50. The class of police cars reveals the second-lowest accuracy score with 97.24% in EfficientNet and 96.66% in ResNet-50. This is mainly because the shapes and colors of both classes are almost identical from the front view. The only difference in appearance is in the back view of the car, where military police cars have the grey or blue color and police cars have a white color. The dominant performance of the ViT-base model can be observed for all classes, especially improving the classification performance for the military police and police cars.

Moreover, the confusion matrices for each classification model are presented in Figure 3. The vertical axis shows the actual labels of corresponding vehicles classes, while the horizontal axis shows the correctly predicted number of vehicles for each class. The confusion matrices shown in Figure 3 indicate the source of the problem with police and military police cars. The ViT model accomplishes this difficult task in contrast to other models.

Furthermore, we have compared the latency for each classification model. The latency is a processing-time measurement to determine the performance of various models for a specific application, where the latency is defined as the time required for processing one image. The ViT model takes 0.017 seconds to classify one image, while EfficientNet and ResNet-50 consume 0.015 seconds and 0.013 seconds, respectively. While the ViT model performs better than other models for the critical vehicle classification task, it reaches a slightly higher latency, which is acceptable.

5 CONCLUSION

This work studies the critical vehicle detection problem in traffic for safety reasons, which is rarely addressed in the literature. First, we have carefully gathered all vehicle classes from traffic surveillance data and various datasets, to generate a large-scale critical vehicle database. The created dataset includes 10,424 images classified into 6 categories: standard cars, fire trucks, ambulances, police cars, military police cars, and dangerous trucks. Second, we have employed the ViT and the EfficientNet and ResNet-50 CNN models to experiment and identify the most suitable model for the task. We have chosen the ViT model because it achieves excellent accuracy for each critical vehicle class. Detailed experimental results show that the ViT-based solution reaches an average accuracy higher than 0.99. Furthermore, experimental latency measurements indicate that all three models have similar values. All the trained models used for the proposed approach are publicly available (Wightman, 2019), (Glenn Jocher, 2021).

ACKNOWLEDGEMENTS

This work is supported by the European ITEA project SMART on intelligent traffic flow systems and Efficient Deep Learning (EDL) RMR project.

REFERENCES

Arinaldi, A., Pradana, J. A., and Gurusinga, A. A. (2018). Detection and classification of vehicles for traffic video analytics. *Procedia computer science*, 144:259–268.

Bochkovskiy, A., Wang, C.-Y., and Liao, H.-Y. M. (2020). Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*.

Carvalho Barbosa, R., Shoaib Ayub, M., Lopes Rosa, R., Zegarra Rodríguez, D., and Wuttisittikulkij, L. (2020). Lightweight pvidnet: A priority vehicles detection network model based on deep learning for intelligent traffic lights. *Sensors*, 20(21):6218.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Dey, R. and Salem, F. M. (2017). Gate-variants of gated recurrent unit (gru) neural networks. In *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*, pages 1597–1600. IEEE.

Dollar, P., Wojek, C., Schiele, B., and Perona, P. (2011). Pedestrian detection: An evaluation of the state of the art. *IEEE transactions on pattern analysis and machine intelligence*, 34(4):743–761.

Girshick, R., Donahue, J., Darrell, T., and Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587.

Glenn Jocher, e. a. (2021). ultralytics/yolov5: v6.0 - YOLOv5n 'Nano' models. https://github.com/ultralytics/yolov5.

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., and Schmidhuber, J. (2016). Lstm: A search space odyssey. *IEEE transactions on neural networks and learning systems*, 28(10):2222–2232.

He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.

Jung, H., Choi, M.-K., Jung, J., Lee, J.-H., Kwon, S., and Young Jung, W. (2017). Resnet-based vehicle classification and localization in traffic surveillance systems. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 61–67.

Kobayashi, T., Hidaka, A., and Kurita, T. (2007). Selection of histograms of oriented gradients features for pedestrian detection. In *International conference on neural information processing*, pages 598–607. Springer.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., and Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.-Y., and Berg, A. C. (2016). Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. (2019). Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32.

Redmon, J. and Farhadi, A. (2018). Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*.

Ren, S., He, K., Girshick, R., and Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28:91–99.

Reynolds, D. A. (2009). Gaussian mixture models. *Encyclopedia of biometrics*, 741:659–663.

Roy, S. and Rahman, M. S. (2019). Emergency vehicle detection on heavy traffic road from cctv footage using deep convolutional neural network. In *2019 International Conference on Electrical, Computer and Communication Engineering (ECCE)*, pages 1–6. IEEE.

Shekade, A., Mahale, R., Shetage, R., Singh, A., and Gadakh, P. (2020). Vehicle classification in traffic surveillance system using yolov3 model. In *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, pages 1015–1019. IEEE.

Simonyan, K. and Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

Suhao, L., Jinzhao, L., Guoquan, L., Tong, B., Huiqian, W., and Yu, P. (2018). Vehicle type detection based on deep learning in traffic scene. *Procedia computer science*, 131:564–572.

Suykens, J. A. and Vandewalle, J. (1999). Least squares support vector machine classifiers. *Neural processing letters*, 9(3):293–300.

Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.

Tan, M., Pang, R., and Le, Q. V. (2020). Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790.

Wang, X., Ma, X., and Grimson, W. E. L. (2008). Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE Transactions on pattern analysis and machine intelligence*, 31(3):539–555.

Wightman, R. (2019). Pytorch image models. https://github.com/rwightman/pytorch-image-models.