Deep Neural Network-based Detection of Road Traffic Objects from Drone-Captured Imagery Focusing on Road Regions

Hoanh Nguyen
Faculty of Electrical Engineering Technology, Industrial University of Ho Chi Minh City, Ho Chi Minh City, Vietnam

***Abstract*—This paper presents a novel deep learning approach for the detection of traffic objects from drone-based imagery, focusing predominantly on the rapid and accurate detection of vehicles within road sections. The proposed method consists of two primary components: a road segmentation module and a vehicle detection network. The former leverages a residual unit with skip-connections to effectively extract road areas, while the latter employs a modified version of the YOLOv3 architecture, tailored for high-accuracy and high-speed vehicle detection. To address the issue of data imbalance, which is a pervasive challenge in drone images, this paper utilizes a range of data augmentation techniques to improve the robustness of the proposed model. Experimental results on the UAVDT and UAVid datasets exhibit that the proposed model attains a substantial boost in accuracy and inference speed of vehicle detection in comparison to the existing methods. These findings underscore the potential of the proposed approach for real-world traffic monitoring applications, where rapid and reliable vehicle detection is paramount.**

Keywords*—Deep learning; drone images; vehicle detection; road segmentation; data imbalance

I. Introduction

As drone technology has rapidly advanced in recent years, numerous practical applications based on images collected from drones have been developed. Among these, the most notable are intelligent processing applications based on images obtained from drones, such as object detection [1-2], object segmentation [3], traffic analysis [4], traffic prediction [5], and work monitoring systems [6]. Compared to ground-collected images, drone-collected images often have many more advantages, such as encompassing information from a vast area, dynamic coverage, and different altitudes and positions. Due to these benefits, processing based on drone images often faces many challenges. These challenges stem from various factors including complex backgrounds, a global perspective, and varying scales of targets. Fig. 1 describes some cases of drone images that pose many challenges for object detection and segmentation tasks. More specifically, objects in drone images are often obscured or overlap with other objects. The number of objects in drone images is usually very large, and the size of the objects in the images is often small.

Given the significant advancements in deep learning, particularly in convolutional neural networks (CNN), numerous methods have been introduced in recent years to address object detection and segmentation using drone images and CNN. In [7], the authors propose an automatic image annotation method, analyze YOLOv3's training behavior on the natural UAVDT dataset, and demonstrate the performance that can be achieved through synthetic training, as well as how synthetic augmentation can enhance the natural training set's performance. Kyrkou et al. [8] present a comprehensive approach to developing a single-shot object detector based on CNN for UAVs, specifically focusing on vehicle detection in resource-constrained environments. The paper covers the entire development process including data collection, training, CNN architecture design, and optimizations for efficient deployment on lightweight embedded processing platforms suitable for drone images. Li et al. [9] introduced the Density-Map guided object detection Network (DMNet) as an inventive approach to tackle the complexities of object detection in high-resolution aerial photos, particularly issues related to vast differences in object size and irregular object distribution. The DMNet, which integrates a density map generation module, an image cropping module, and an object detector, uses pixel intensity to establish a subtle boundary for image cropping and to discern object scales. In [10], the authors propose a separate resampling algorithm to alter the input test images' size and subsequently extend the object's impact in deeper layers of the detection model. They utilize a pre-trained Faster R-CNN [11] object detection model with Inception-V2 [12], applying transfer learning to submeter satellite images with passenger vehicles as the target objects. In [13], the author present a novel vision-based system for vehicle detection and counting on highways, aiming to address the challenge of detecting vehicles of varying sizes. This system utilizes a novel segmentation technique to partition the highway road surface in the image into distant and near areas. Subsequently, it leverages the YOLOv3 network for vehicle detection. Recently, Feng et al. [14] suggested utilizing the mean classification score as a metric for gauging the classification accuracy of each category during training. They introduced the Equilibrium Loss (EBL) and Memory-augmented Feature Sampling (MFS) techniques to ensure balanced classification. Together, EBL and MFS notably enhance detection performance for less represented classes while either preserving or boosting performance for the more prevalent ones. In addition to the methods mentioned above, references [15-18] provide systematic reviews of object detection and segmentation methods based on drone images.

While the aforementioned methods have achieved certain successes, there remain numerous issues that need to be

addressed to construct an effective model for object detection based on drone images. This paper introduces a proficient model for detecting objects in drone images. Aiming to provide efficient data for intelligent traffic monitoring applications, the model proposed in this paper performs vehicle object detection based on regions of interests (RoIs) rather than on the entire image. Detecting objects based on RoIs helps to eliminate unrelated areas during processing, not only increasing the model's accuracy but also significantly enhancing execution speed, particularly for high-resolution images obtained from drones. To efficiently create RoIs, specifically road sections, this paper proposes applying a segmentation model for road detection. Based on extracted road sections, a method is proposed to enhance both the accuracy and inference speed of vehicle detection from road sections. Moreover, in response to the widespread issue of data imbalance often encountered in drone imagery, this paper applies an assortment of data augmentation strategies aimed at enhancing the resilience and reliability of the proposed model. Finally, this paper also proposes using suitable models and datasets that meet the requirements.

The paper is organized as follows: Section II delves into the details of the proposed methodology. Section III presents the results derived from the framework's implementation. Finally, Section IV concludes the paper and highlights potential avenues for future research.

Fig. 1. Some images illustrate the challenges of object detection and segmentation tasks with images from drones.

II. Methodology

A. Overview of the Proposed Method

Fig. 2 illustrates the overall structure of the model proposed for the problem of road traffic object detection from drone images. Aiming at detecting objects on the road, specifically vehicles, with high inference speed to provide real-time information for road management systems, a deep learning network is first used for road detection and segmentation. Extracting the road sections helps the model focus on detecting objects on the road, thereby not only significantly increasing the inference speed of the overall model but also improving accuracy. The input to this deep learning network is the input images, and the output is the predicted road sections. Based on the extracted road sections, a deep learning network based on the YOLOv3 architecture is designed for object detection, specifically vehicles on the road. Using a vehicle detection model based on the YOLOv3 architecture significantly improves the model's inference time, especially with images obtained from drones where the number of objects on the road is considerable. Additionally, the paper also proposes using data augmentation strategies to address issues related to data imbalance often encountered in drone imagery. Details about each proposed network will be presented in the following sections.

B. Road Segmentation

This paper approaches road segmentation in images as a binary segmentation task, classifying each pixel in the input image as either part of the road or the background. Several models have been proposed for segmentation, such as UNet [19], Segnet [20], DeepLabv3+ [21], or the more recent DoubleUNet [22]. These models typically combine an encoder for feature extraction with a decoder for segmentation. The encoder is critical in extracting features, capturing contextual information, reducing dimensionality, creating a hierarchical representation, and making use of transfer learning. Conversely, the decoder is responsible for upsampling, reconstructing the feature maps, refining the segmentation output with contextual information, applying non-linear mappings, and generating the final segmentation output through multi-level feature fusion and skip connections. In pursuit of high accuracy and fast inference speed, this paper proposes the use of the ResNet50 model [23] as the encoder and a combination of residual blocks and other operations [24] as the decoder, as depicted in Fig. 3. Specifically, this paper utilizes the ResNet50 model, which is pre-trained on the ImageNet dataset [25], to ensure smoother convergence and elevate the overall performance. The decoder consists of four blocks, each including upsampling, concatenation, and skip-connections operations. In detail, each block's input feature map is initially upsampled to a higher resolution using transpose convolution. Following this, a concatenation operation merges the upsampled feature map with its encoder counterpart. A residual unit with skip-connections then produces the final feature map for that block. Opting for residual blocks over standard convolutional ones streamlines network training and guarantees undegraded information propagation due to the skip connections. The decoder's final block output undergoes a 1×1 convolution layer and a sigmoid function, resulting in the output segmentation map.

For training the road segmentation network, this paper employs Dice Loss [26] as the main loss function. Dice Loss is particularly useful in segmentation tasks where the classes are imbalanced. The goal of the road segmentation task is to categorize each image pixel as either road or background. Given that the number of pixels associated with roads is typically far less than those tied to the background, this presents a significantly imbalanced problem. Dice Loss helps mitigate this issue by maintaining a balance between the foreground and the background. Mathematically, the Dice Loss can be defined as:

$$L_d = 1 - 2\frac{y_{pre} \cdot y_{gt} + \varepsilon}{y_{pre} + y_{gt} + \varepsilon} \quad (1)$$

where y_{pre} and y_{gt} represent the predicted and ground truth, respectively. ε is a minute positive quantity employed to prevent a division by zero issue.

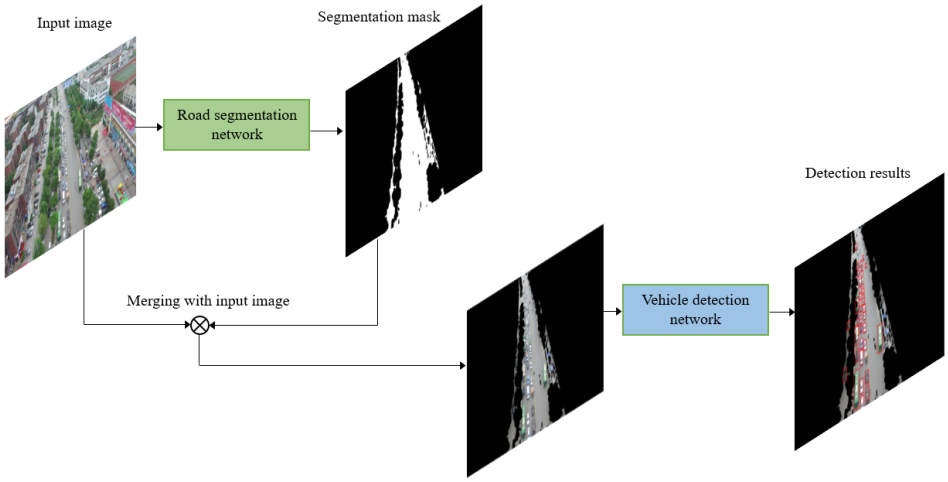

Fig. 2. The structure of the proposed approach.

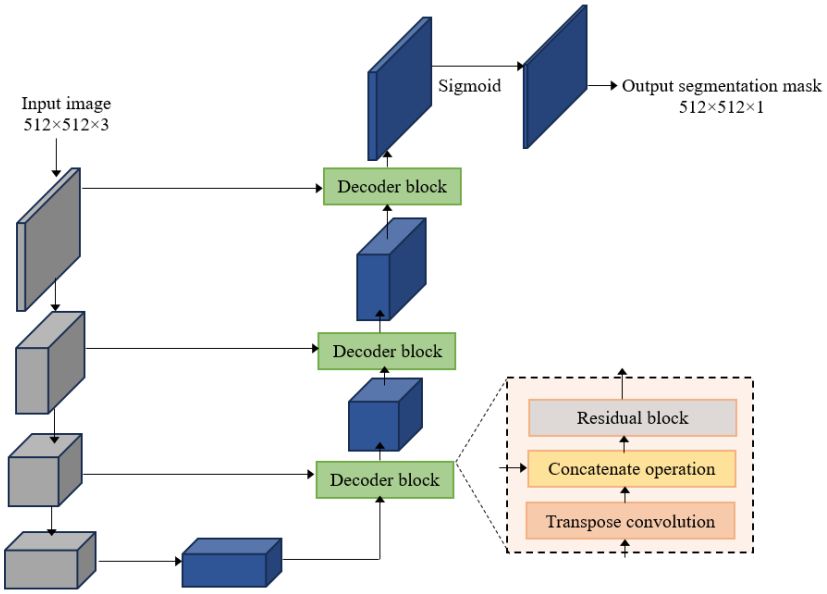

Fig. 3. Encoder-decoder structure for road segmentation.

C. *Vehicle Detection based on Road Sections*

Aiming for quick and accurate vehicle detection based on road sections extracted from input images, especially for small vehicles, this paper proposes a vehicle detection model based on the YOLOv3-tiny architecture [27]. Specifically, modifications were made to the YOLOv3-tiny model based on two criteria: model size and its capability to detect small objects. In CNN architecture, deeper layers containing larger numbers of channels and smaller sizes typically store rich semantic information, beneficial for object classification. Conversely, shallower layers with fewer channels and larger sizes typically house rich spatial information, useful for preserving object structure details. Since vehicle detection task only distinguishes between vehicles and background classes, it is significantly simpler than generic object detection. As a result, the shallower network layers can be reduced in the number of channels to decrease the model's complexity without impacting its accuracy. Based on these analyses, this paper implemented changes to the structure of the first convolutional layers of the YOLOv3-tiny architecture. Specifically, the number of filters in the first two convolutional layers was reduced to three. The rest of the convolutional layers maintained their original number of filters. Table I details the structure of the proposed model in this paper and the original YOLOv3-tiny model. The detection head makes predictions on two feature maps with scales of 13×13 and 26×26. With these modifications, the proposed model can significantly reduce computation costs while maintaining network performance. Additionally, the filter size in the detection layers was changed from 3×3 to 1×1, improving the model's nonlinearity and aiding the detection model in learning difficult samples. With these changes, the new model has reduced FLOPs to 2.5BFLOPs compared to the original model's 5.4BFLOPs, while the model size has shrunk to 20MB compared to the original 34MB.

TABLE I. COMPARING THE STRUCTURE OF THE ORIGINAL YOLOV3-TINY MODEL AND THE MODEL PROPOSED IN THE PAPER

Layer	Original YOLOv3-tiny			Proposed architecture		
	Type	Filter	Output	Type	Filter	Output
0	Convolutional	3×3×16	416×416×16	Convolutional	3×3×3	416×416×3
1	Max Pooling	2×2	208×208×16	Max Pooling	2×2	208×208×3
2	Convolutional	3×3×32	208×208×32	Convolutional	3×3×3	208×208×3
3	Max Pooling	2×2	104×104×32	Max Pooling	2×2	104×104×3
4	Convolutional	3×3×64	104×104×64	Convolutional	3×3×64	104×104×64
5	Max Pooling	2×2	52×52×64	Max Pooling	2×2	52×52×64
6	Convolutional	3×3×128	52×52×128	Convolutional	3×3×128	52×52×128
7	Max Pooling	2×2	26×26×128	Max Pooling	2×2	26×26×128
8	Convolutional	3×3×256	26×26×256	Convolutional	3×3×256	26×26×256
9	Max Pooling	2×2	13×13×256	Detection		
10	Convolutional	3×3×512	13×13×512	Max Pooling	2×2	13×13×256
11	Convolutional	1×1×256	13×13×256	Convolutional	3×3×512	13×13×512
12	Convolutional	3×3×255	13×13×255	Detection		
13	Detection					

D. Data Augmentation Strategy

Since data imbalance among classes presents a significant challenge for vision tasks based on drone images, this paper proposes several data augmentation techniques to address this issue. Fig. 4 displays the outcomes of the data augmentation techniques applied in this paper on a consistent input image. The strategies employed to augment drone image data in this study encompass random erasing, random rotation, random brightness, random cropping, and random zoom.

1) Random erasing: Random erasing [28] involves selecting a rectangular area within an image at random and replacing its pixels with arbitrary values. This region is determined using a uniform distribution, with both the area and aspect ratios chosen randomly. When parts of the input image are randomly erased during training, it compels the model to develop more adaptable and robust representations. This means the model has to identify the correct class without depending solely on the complete image, enhancing its focus on pertinent sections of the input. This can enhance the model's generalization capabilities, potentially leading to superior performance on unfamiliar data.

2) Random rotation: Random rotation is achieved by rotating the image by a random degree between -90 and +90 degrees. By randomly rotating the image, the model is encouraged to learn to recognize the object in different orientations. This makes the model more robust to the orientation of objects in the input data. Let (x, y); (x', y') be the coordinates of the bounding boxes before and after implementing random rotation. In that case,

$$\begin{cases} x' = (x - M_x).cos\alpha - (y - M_y).sin\alpha + M_x \\ y' = (x - M_x).sin\alpha - (y - M_y).cos\alpha + M_y \end{cases} \quad (2)$$

where α is rotation angle and (M_x, M_y) is the center coordinate of the input image.

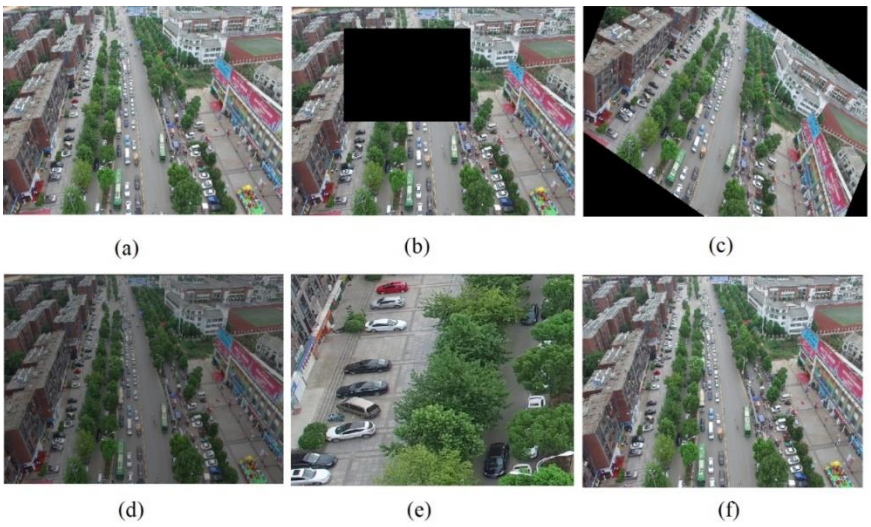

Fig. 4. Data augmentation used in this paper. (a) Original image; (b) Random erasing; (c) Random rotation; (d) Random brightness; (e) Random crop; (f) Random zoom.

3) Random brightness: Random brightness involves adjusting the brightness of an image by a random factor. It's usually achieved by converting the image to the HSV color space, adding a random value to the *V* (Value) channel, and then converting back to the original color space. By modifying the brightness of the image, the model can be trained to be invariant to different lighting conditions. This means that the trained model can recognize an object or feature in an image regardless of whether the image is bright, normally lit, or dim.

4) Random crop: Random crop involves selecting a random subsection of the input image for training. The cropped region is smaller than the original image and is resized to the input dimensions of the model. By training the model on a diverse set of cropped images, it can learn to focus on different parts of an object and recognize an object even if only part of it is visible. It can also help to mitigate overfitting as the model cannot rely on the position of features in the image.

5) Random zoom: Random zoom involves randomly zooming into or out of an image by a certain amount. This is typically done by resizing the image (upscaling or downscaling) and then cropping or padding it to match the original dimensions. By randomly zooming in or out, the model can learn to recognize objects or features at different scales. It makes the model more scale-invariant, which is beneficial when objects in the test data may appear at different sizes than in the training data. The coordinates of the bounding boxes are updated after implementing random zoom as follows:

$$\begin{cases} x' = \frac{w}{2} + \frac{(x - \frac{w}{2})}{R} \\ y' = \frac{h}{2} + \frac{(y - \frac{h}{2})}{R} \end{cases} \quad (3)$$

where R is zoom ration and (w, h) is the width and height of the input image.

III. Results

A. Dataset

This paper utilizes distinct datasets for different tasks as specified in Table II. Specifically, the UAVDT dataset [29] is used to train the vehicle detection network. This dataset is tailored for vehicle detection and tracking tasks, encompassing three categories: car, truck, and bus. For the vehicle detection evaluation in this study, all classes are grouped under a singular category termed 'vehicle'. The images feature a resolution of 1080×540 pixels and capture diverse typical scenes, including squares, arterial roads, and toll stations. For training the road segmentation network, the UAVid dataset [30] is employed. UAVid consists of 300 drone images with resolutions of 4096×2160 or 3840×2160 pixels, captured at a slanted angle, enhancing the intricacy and scale variance of urban street scenes with complex foreground-background elements. Given the substantial image sizes, this paper derives 10,000 random, non-overlapping 512×512 patches from the UAVid dataset. Of these, 8,000 are designated as the training set, while the remaining 2,000 are split equally between the validation and testing sets. For the purpose of road segmentation, only the annotations relevant to roads are used for training and evaluation in the road segmentation network. Additionally, the UAVid dataset is also used for a joint evaluation of the proposed model. In this evaluation setting, only road annotations are employed throughout the paper, and images lacking road annotations are excluded from the dataset. Furthermore, this study manually uses all vehicle annotations within road sections for object detection training and evaluation.

TABLE II. Datasets used in this Paper

Dataset	Road segmentation	Vehicle detection	Joint segmentation and detection
UAVDT [29]		√	
UAVid	√		√

B. Implementation Details

All road segmentation and vehicle detection networks are trained on a NVIDIA RTX 4080 GPU with the support of the PyTorch library. For the road segmentation network, the ResNet50 model, pretrained on the ImageNet dataset, is used as the baseline encoder. This enhances the precision of feature extraction, consequently improving segmentation performance. To boost training performance, an initial learning rate of $1e^{-4}$ is used to update the parameters, which is then reduced to $1e^{-7}$ after six consecutive epochs to achieve a better loss rate. The Adam optimizer [31] is utilized to fine-tune the model. The model is trained over 20 epochs with a batch size of 16. For the vehicle detection network, training is carried out using default configurations with a few minor modifications. More specifically, the DarkNet model [27] is deployed, and the SGD optimizer with momentum and weight decay factors of 0.9 and 0.001 respectively is used in the detector training process. The vehicle detection model is trained for 100 epochs with a batch size of 32. A step learning schedule is also employed to gradually reduce the learning rate.

C. Road Segmentation Results

This paper conducts experiments with various models to evaluate the effectiveness of the proposed model for the road segmentation task. The compared models are based on EfficientNet [32] and MobileNetv2 [33] as encoders, while the decoders are networks such as DeepLabV3, FPN [34], and Unet. Experiments are performed on the same UAVid dataset with identical training and testing sets. Fig. 5 illustrates the results of the segmentation models used in the experiments, including the model proposed in this paper. It can be seen that the proposed model achieves the best inference speed, while maintaining accuracy comparable to the other models. In terms of accuracy, the DeepLabV3 with EfficientNet model performs the best. However, this model requires 8.2ms as inference time, which is not suitable for intelligent transportation systems requiring real-time processing. Additionally, the results in Fig. 5 also show that models using complex decoder structures, with many layers like DeepLabV3, often require longer processing times due to higher computational demands. The comparison results show that designing an encoder-decoder model that leverages a residual unit with skip-connections, as

proposed in this paper, is very effective both in terms of accuracy and inference speed for the road segmentation task.

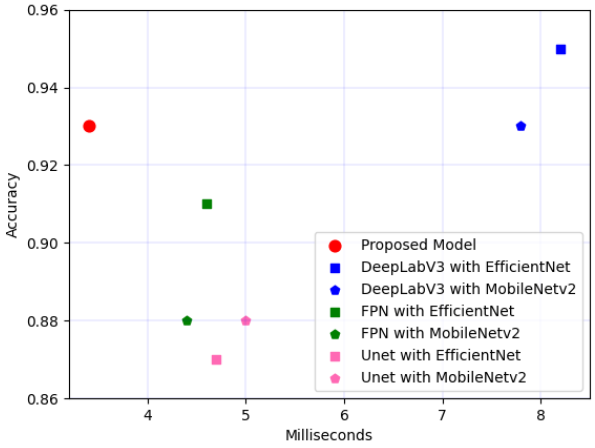

Fig. 5. Performance of different models on the UAVid dataset.

D. Vehicle Detection Results

To evaluate the vehicle detection network proposed in this paper, several models have been tested on the UAVDT dataset, including YOLOv3, YOLOv3-tiny, YOLOv4 [35], and SSD-MobileNet [36]. Table III presents the vehicle detection performance of various models on the UAVDT dataset, with the metrics being the Average Precision (AP) in percentage and the speed in milliseconds. From the results, it is clear that YOLOv4 exhibits the highest AP of 86.4, closely followed by YOLOv3 with 82.1. However, when it comes to speed, YOLOv4 falls short with a processing time of 10.4ms, compared to YOLOv3's 12.2ms. On the other hand, YOLOv3-tiny, known for its lightweight architecture, posts a decent AP of 69.4 but shines in speed with a processing time of 6.4ms. The proposed model, a modification of the YOLOv3-tiny architecture, was designed specifically to optimize the model size and improve detection of small objects. Despite achieving a slightly lower AP of 76.2 compared to the original YOLOv3 models, it considerably outperforms all other models in terms of speed with an impressive 4.2ms. This result reflects the efficiency of the proposed design in balancing both precision and speed. In comparison to SSD-MobileNet, which has an AP of 80.4 and speed of 10.6ms, the proposed model excels in inference speed, showing its potential for real-time applications. Therefore, the proposed model offers a promising approach for traffic monitoring, where speed and accurate vehicle detection is crucial.

TABLE III. VEHICLE DETECTION PERFORMANCE OF DIFFERENT MODELS ON THE UAVDT DATASET

Models	AP (%)	Speed (ms)
YOLOv3	82.1	12.2
YOLOv3-tiny	69.4	6.4
YOLOv4	86.4	10.4
SSD-MobileNet	80.4	10.6
Proposed model	76.2	4.2

E. Joint Evaluation Results

For joint evaluation, the road segmentation and vehicle detection networks have been integrated to carry out the task of vehicle detection within road sections. Based on the UAVid dataset, labels have been modified to include only vehicles in road sections to determine how accurately the combined model can detect vehicles in these areas. Table IV presents the overall results of several models, including Unet + YOLOv3 and Unet + YOLOv3-tiny. In Table IV, two parts of the comparison are introduced, which include the detection of vehicles in road sections and the detection of vehicles across the entire image. It can be seen that the detection of vehicles in road sections significantly improves the accuracy and inference speed of all models compared to vehicle detection across the entire image. This can be explained by the fact that by focusing only on the necessary parts of the image during detection, the computational cost and the number of objects that need to be predicted are substantially reduced. These findings suggest that intelligent transportation applications could leverage these results to build more efficient systems in their design, thereby facilitating easier system development.

TABLE IV. JOINT EVALUATION RESULTS ON THE UAVID DATASET

Models	AP (%)		Speed (ms)	
	Road sections	Entire image	Road sections	Entire image
Unet + YOLOv3	76.4	62.8	11.6	16.2
Unet + YOLOv3-tiny	62.1	54.4	8.4	10.3
Proposed model	74.1	60.5	6.9	8.4

IV. CONCLUSIONS

This paper has designed a novel deep learning method for the detection of traffic objects from drone-based imagery, specifically focusing on the rapid and accurate detection of vehicles within road sections. The proposed method consists of two key components: a road segmentation network and a vehicle detection network. The segmentation network utilizes a residual unit with skip-connections to effectively predict road areas, while the vehicle detection network leverages a modified version of the YOLOv3 architecture, fine-tuned for high-accuracy and high-speed vehicle detection. Moreover, this study addressed the challenge of data imbalance inherent in drone images by implementing various data augmentation techniques, thereby enhancing the model's robustness. The experimental results achieved on the UAVDT and UAVid datasets highlighted the effectiveness of the proposed model. It not only enhanced the accuracy of vehicle detection but also improved the inference speed as compared to existing methods. These results highlight the potential of the proposed approach for practical traffic monitoring applications, where rapid and accurate vehicle detection is of utmost importance. However, it's important to note that the proposed model's effectiveness may be limited to adverse weather conditions or low-light scenarios, as it heavily relies on visual data captured by drones, which can be adversely affected by such factors. Additionally, the model's performance might degrade when applied to highly congested traffic scenes with overlapping vehicles, posing challenges in accurate object detection. For future work, this

paper plans to extend this approach to the detection of more diverse traffic objects beyond vehicles, such as pedestrians and cyclists.

REFERENCES

[1] Patrik, Aurello, Gaudy Utama, Alexander Agung Santoso Gunawan, Andry Chowanda, Jarot Sembodo Suroso, and Widodo Budiharto. "Modeling and implementation of object detection and navigation system for quadcopter drone." *ICIC Express Letters* 13, no. 6 (2019): 461-468.

[2] Arrahmah, Annisa Istiqomah, Rissa Rahmania, and Dany Eka Saputra. "Comparison between convolutional neural network and K-nearest neighbours object detection for autonomous drone." *Bulletin of Electrical Engineering and Informatics* 11, no. 4 (2022): 2303-2312.

[3] Eerapu, Karuna Kumari, Shyam Lal, and A. V. Narasimhadhan. "O-SegNet: Robust encoder and decoder architecture for objects segmentation from aerial imagery data." *IEEE Transactions on Emerging Topics in Computational Intelligence* 6, no. 3 (2021): 556-567.

[4] Liu, Shuai, Xin Li, Huchuan Lu, and You He. "Multi-object tracking meets moving UAV." In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8876-8885. 2022.

[5] Yin, Xueyan, Genze Wu, Jinze Wei, Yanming Shen, Heng Qi, and Baocai Yin. "Deep learning on traffic prediction: Methods, analysis, and future directions." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 6 (2021): 4927-4943.

[6] Casierra, Cristian Benjamín García, Carlos Gustavo Calle Sánchez, Javier Ferney Castillo García, and Felipe Muñoz La Rivera. "Methodology for Infrastructure Site Monitoring using Unmanned Aerial Vehicles (UAVs)." *International Journal of Advanced Computer Science and Applications* 13, no. 3 (2022).

[7] Krump, Michael, Martin Ruß, and Peter Stütz. "Deep learning algorithms for vehicle detection on UAV platforms: first investigations on the effects of synthetic training." In *Modelling and Simulation for Autonomous Systems: 6th International Conference, MESAS 2019, Palermo, Italy, October 29–31, 2019, Revised Selected Papers 6*, pp. 50-70. Springer International Publishing, 2020.

[8] Kyrkou, Christos, George Plastiras, Theocharis Theocharides, Stylianos I. Venieris, and Christos-Savvas Bouganis. "DroNet: Efficient convolutional neural network detector for real-time UAV applications." In *2018 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pp. 967-972. IEEE, 2018.

[9] Li, Changlin, Taojiannan Yang, Sijie Zhu, Chen Chen, and Shanyue Guan. "Density map guided object detection in aerial images." In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, pp. 190-191. 2020.

[10] Mansour, Ahmad, Wessam M. Hussein, and Ehab Said. "Small objects detection in satellite images using deep learning." In *2019 Ninth International Conference on Intelligent Computing and Information Systems (ICICIS)*, pp. 86-91. IEEE, 2019.

[11] Ren, Shaoqing, Kaiming He, Ross Girshick, and Jian Sun. "Faster r-cnn: Towards real-time object detection with region proposal networks." *Advances in neural information processing systems* 28 (2015).

[12] Szegedy, Christian, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. "Rethinking the inception architecture for computer vision." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 2818-2826. 2016.

[13] Song, Huansheng, Haoxiang Liang, Huaiyu Li, Zhe Dai, and Xu Yun. "Vision-based vehicle detection and counting system using deep learning in highway scenes." *European Transport Research Review* 11, no. 1 (2019): 1-16.

[14] Feng, Chengjian, Yujie Zhong, and Weilin Huang. "Exploring classification equilibrium in long-tailed object detection." In *Proceedings of the IEEE/CVF International conference on computer vision*, pp. 3417-3426. 2021.

[15] Bisio, Igor, Chiara Garibotto, Halar Haleem, Fabio Lavagetto, and Andrea Sciarrone. "A systematic review of drone based road traffic monitoring system." *IEEE Access* (2022).

[16] Chen, Jing, Qichao Wang, Harry H. Cheng, Weiming Peng, and Wenqiang Xu. "A review of vision-based traffic semantic understanding in ITSs." *IEEE Transactions on Intelligent Transportation Systems* (2022).

[17] Zhang, Xingchen, Yuxiang Feng, Panagiotis Angeloudis, and Yiannis Demiris. "Monocular visual traffic surveillance: A review." *IEEE Transactions on Intelligent Transportation Systems* 23, no. 9 (2022): 14148-14165.

[18] Wang, Wenguan, Tianfei Zhou, Fatih Porikli, David Crandall, and Luc Van Gool. "A survey on deep learning technique for video segmentation." *arXiv e-prints* (2021): arXiv-2107.

[19] Ronneberger, Olaf, Philipp Fischer, and Thomas Brox. "U-net: Convolutional networks for biomedical image segmentation." In *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2015: 18th International Conference, Munich, Germany, October 5-9, 2015, Proceedings, Part III 18*, pp. 234-241. Springer International Publishing, 2015.

[20] Badrinarayanan, Vijay, Alex Kendall, and Roberto Cipolla. "Segnet: A deep convolutional encoder-decoder architecture for image segmentation." *IEEE transactions on pattern analysis and machine intelligence* 39, no. 12 (2017): 2481-2495.

[21] Chen, Liang-Chieh, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. "Encoder-decoder with atrous separable convolution for semantic image segmentation." In *Proceedings of the European conference on computer vision (ECCV)*, pp. 801-818. 2018.

[22] Jha, Debesh, Michael A. Riegler, Dag Johansen, Pål Halvorsen, and Håvard D. Johansen. "Doubleu-net: A deep convolutional neural network for medical image segmentation." In *2020 IEEE 33rd International symposium on computer-based medical systems (CBMS)*, pp. 558-564. IEEE, 2020.

[23] He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770-778. 2016.

[24] Zhang, Zhengxin, Qingjie Liu, and Yunhong Wang. "Road extraction by deep residual u-net." *IEEE Geoscience and Remote Sensing Letters* 15, no. 5 (2018): 749-753.

[25] Deng, Jia, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. "Imagenet: A large-scale hierarchical image database." In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248-255. Ieee, 2009.

[26] Sudre, Carole H., Wenqi Li, Tom Vercauteren, Sebastien Ourselin, and M. Jorge Cardoso. "Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations." In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, pp. 240-248. Springer International Publishing, 2017.

[27] Redmon, Joseph, and Ali Farhadi. "Yolov3: an incremental improvement. 2018." *arXiv preprint arXiv:1804.02767* 20 (1804).

[28] Zhong, Zhun, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. "Random erasing data augmentation." In *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 07, pp. 13001-13008. 2020.

[29] Du, Dawei, Yuankai Qi, Hongyang Yu, Yifan Yang, Kaiwen Duan, Guorong Li, Weigang Zhang, Qingming Huang, and Qi Tian. "The unmanned aerial vehicle benchmark: Object detection and tracking." In *Proceedings of the European conference on computer vision (ECCV)*, pp. 370-386. 2018.

[30] Lyu, Ye, George Vosselman, Gui-Song Xia, Alper Yilmaz, and Michael Ying Yang. "UAVid: A semantic segmentation dataset for UAV imagery." *ISPRS journal of photogrammetry and remote sensing* 165 (2020): 108-119.

[31] Kingma, Diederik P., and Jimmy Ba. "Adam: A method for stochastic optimization." *arXiv preprint arXiv:1412.6980* (2014).

[32] Tan, Mingxing, and Quoc Le. "Efficientnet: Rethinking model scaling for convolutional neural networks." In *International conference on machine learning*, pp. 6105-6114. PMLR, 2019.

[33] Sandler, Mark, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. "Mobilenetv2: Inverted residuals and linear bottlenecks." In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510-4520. 2018.

[34] Lin, Tsung-Yi, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. "Feature pyramid networks for object detection." In *Proceedings*.

[35] Bochkovskiy, Alexey, Chien-Yao Wang, and Hong-Yuan Mark Liao. "Yolov4: Optimal speed and accuracy of object detection." *arXiv preprint arXiv:2004.10934* (2020).

[36] Liu, Wei, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. "Ssd: Single shot multibox detector." In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pp. 21-37. Springer International Publishing, 2016.