Motorcyclist Helmet Violation Detection Framework by Leveraging Robust Ensemble and Augmentation Methods

Thien Van Luong[2*] Huu Si Phuc Nguyen[1] Duy Khanh Dinh[1] Viet Hung Duong[1]
Duy Hong Sam Vo[1] Huan Vu[3] Minh Tuan Hoang[1] Tien Cuong Nguyen[1]
[1] VNPT AI, VNPT Group, Ha Noi, Viet Nam
[2] Faculty of Computer Science, Phenikaa University, Ha Noi, Viet Nam
[3] University of Transport and Communications, Hanoi, Vietnam
[2]thien.luongvan@phenikaa-uni.edu.vn, [3]huan.vu@utc.edu.vn
{[1]phucnhs,[1]khanhdd,[1]hungdv,[1]samvdh,[1]tuanhgm,[1]nguyentiencuong}@vnpt.vn

Abstract

Traffic Monitoring Systems play a crucial role in real-life scenarios by improving traffic flow and reducing violations. Among these violations, helmet non-compliance is particularly common in countries where motorcycles are the primary mode of transportation. However, deploying an automatic violation capturing for helmet non-compliance in the real world presents challenges due to diverse traffic situations, environmental conditions, varying object sizes, and severely imbalanced datasets. In order to address these challenges, we propose a novel deep learning framework for helmet violation detection, which consists of four different modules, namely, object detection, object association, post-processing for tracking, and score correction. In particular, we develop a robust ensemble method to take advantage of various state-of-the-art object detection models such as YOLOv7, YOLOv8, Co-DETR, and EfficientDet. Furthermore, to address the issue of data imbalance, we propose two copy and paste data augmentation techniques for enriching data samples of rare classes. As a result, our approach yields a substantial 7.43% mAP enhancement over the baseline Co-DETR model, achieving a final score of 0.4792 in the 2024 AI City Challenge Track 5 test set and ranking 3rd among the competing teams.

1. Introduction

Motorcycles are ubiquitous on roads across many countries, serving as a popular mode of transportation. However, the alarming number of traffic accidents involving motorcycles remains a pressing concern. Among the various safety measures, helmets play a pivotal role in protecting both riders and passengers. Therefore, ensuring helmet compliance is essential for reducing head injuries and preventing fatalities.

A human-based monitoring system requires massive resources, and it is difficult to handle a large number of streaming videos simultaneously. The demand for automatic violation capturing has recently increased to solve the problem of the labor shortage for manual monitoring, impose law enforcement, and improve transparency. In recent years, there have been a number of research works [9, 22], which successfully addressed the problems of density estimation, speed estimation, vehicle classification, and violation detection using deep learning approaches.

The recent developments in computer vision and deep learning, especially in object detection and segmentation such as SSD [21], YOLO series [25], Segment Anything [17], and multiple object tracking algorithms such as SORT [3] and Deep SORT [36] make automated helmet detection more feasible in real-world deployment. Based on these fundamental techniques, various systems have been developed to monitor motorcycles and identify violations related to helmet usage. For example, recent research [5, 7, 30] has demonstrated the effectiveness of automated helmet violation detection systems. These systems pave the way for our study to address the issue of helmet law violations.

The 2024 AI City Challenge Track 5 [35] offers a challenging dataset that contains various environmental conditions and camera angles, different traffic situations, and the similarity of object types. The traditional object detection and classification pipeline finds it difficult to solve this complex issue. In this paper, we propose a novel framework that combines different object detection models, tracking algorithms, creative data augmentation, and model ensemble methods in order to improve detection accuracy. In particular, our main contributions are summarized as follows:

- We propose a novel deep learning framework for hel-

*Corresponding author.

met violation detection, which consists of four modules, namely, object detection, object association, post-processing for tracking, and score correction (see Figure 1). In particular, the object detection module includes two sub-modules, namely, head detection and person and vehicle detection, where we apply the Weighted Boxes Fusion [28] ensemble method to combine four state-of-the-art object detection models, such as, YOLOv7 [34], YOLOv8 [16], Co-DETR [38], and EfficientDet [29]. Then, the detected head, person, vehicle objects are fed into the object association module to assign all possible pairs of human-motorbike and human-head by using the same tracking ID for them.

- Next, we develop the post-processing for tracking module relying on SORT [3] and modified Kalman Filter Estimation to detect the vehicle direction information, which is then utilized for detecting possible Passenger 0 and Passenger 2 objects (see Table 1 for these classes). In order to address the issue of data imbalance, we further develop the score correction module, which implements a suitable offset value to confidence scores of Passenger 0 and 2.
- Last but not least, in order to further mitigate the impact of data imbalance, we propose two copy and paste augmentation methods, namely, manual augmentation and automatic augmentation, which aim to generate additional data samples of rare classes such as Passenger 2 without helmet and Passenger 0 without helmet.
- As a result, thanks to all these proposed techniques in our framework, particularly data augmentation, ensemble, and score correction methods, we managed to significantly improve the detection performance by 7.43% mAP over the Co-DETR baseline, which does not include any of these techniques. Specifically, we achieved a final score of 0.4792 in the 2024 AI City Challenge Track 5 test set and ranked 3rd among the competing teams.

The rest of the paper is organized as follows. Section 2 provides a review of existing works on object detection, helmet detection, multi-object tracking, and data augmentation. Section 3 presents our proposed framework for helmet violation detection. The experiment results are analyzed in Section 4. Finally, Section 5 concludes the paper.

2. Related Work

2.1. Helmet Detection for Motorcyclists

In last year's challenge, some fascinating solutions have been proposed; for example, [33] relies on a voting mechanism-based tracking to reduce label switching, [30, 31] focus on improving object detection algorithm, [5, 7] improve the passenger recall and perform the label refinement using tracking algorithm, and [1] concentrates on the environment factors, data sampling and augmentation. Note that most of the aforementioned solutions employ object detection algorithms as a baseline. Additionally, [30] uses an addition detection model for cropped images, while the tracking algorithms are widely adopted by [5, 7, 31] for the label correction and refinement.

2.2. Object Detection

Object detection is a long-history topic in computer vision. Handcrafted feature techniques often rely on specialized methods like Histogram of Gradient [6] or Local Binary Patterns [2] to extract meaningful features from images. Once these features are extracted, they serve as the foundation for classification tasks. Based on the extracted features, well-known classifiers such as Support Vector Machine [14] or K-nearest neighbor [24] can be highly applicable for classifying objects. In the deep learning era, the landscape of object detection algorithms has witnessed exponential growth and advancement. They are divided into two main categories: two-stage object detection, such as Fast-RCNN [11] and Mask-RCNN [13] and one-stage object detection, such as SSD [21] and YOLO series [25]. In recent years, some transformer-based object detection algorithms such as Detecting Objects with Transformers (DETR) [4] and Co-DETR [38] archive the state-of-the-art object detection performance on COCO dataset [20].

2.3. Multiple Object Tracking

Multiple Object Tracking (MOT) [19] is an important task in computer vision that detects and associates objects in consecutive frames. The main aim is to recognize and locate objects of interest in each frame and then link them across frames to keep track of their movements over time. However, this is a very challenging task due to the object occlusion and diverse environments. In order to tackle such challenges, the MOT algorithms usually combine object detection and data association techniques. Particularly, in the detection phase, objects are identified in individual frames, while in the tracking phase, these detected objects are linked across frames to create trajectories.

Simple Online Realtime Tracking (SORT) [3] is one of the most popular MOT algorithms and thus is regarded as a straightforward baseline for MOT problems. In particular, SORT efficiently associates detection results with Kalman filter predictions. Furthermore, the simplicity of SORT makes it appropriate for real-time applications. Recently, several works have adapted tracking algorithms to improve detection results by carrying out the label correction and refinement [5, 7] in 2023 AI City Challenge Track 5 [23]. Note that in [5], the authors employed a tracking algorithm for the label refinement of P2 objects, while the authors in [7] utilized a modified version of SORT for the label correction. Here, P2 objects refer to Passenger 2 on the motorbike, as defined in Table 1.

Table 1. Annotation details of 2024 AI City Challenge Track 5 [35]

Class ID	Class Name	Description	Object Type
1	Motorbike	Motorcycle	Motorbike
2	DHelmet	Motorcycle driver, wearing a helmet	Driver
3	DNoHelmet	Motorcycle driver, not wearing a helmet	Driver
4	P1Helmet	Passenger 1 of the motorcycle, wearing a helmet	P1
5	P1NoHelmet	Passenger 1 of the motorcycle, not wearing a helmet	P1
6	P2Helmet	Passenger 2 of the motorcycle, wearing a helmet	P2
7	P2NoHelmet	Passenger 2 of the motorcycle, not wearing a helmet	P2
8	P0Helmet	Child sitting in front of the Driver, wearing a helmet	P0
9	P0NoHelmet	Child sitting in front of the Driver, not wearing a helmet	P0

2.4. Data Augmentation

In the field of computer vision, data augmentation plays a crucial role in improving the object detection performance. Particularly, data augmentation helps expand the training datasets by generating new samples. This is significantly important in tackling imbalanced datasets. For example, data augmentation techniques such as Augmix [15] and CutMix [37] have shown substantial improvements in object detection performance. In addition, for the problem of the limited dataset, Generative Adversarial Network (GAN) [12] can be exploited to generate synthetic data. For instance, [32] employs Bidirectional GAN to generate synthetic motorbike data samples. Another interesting data augmentation technique, namely, Copy and Paste Augmentation (CPA) [10], randomly selects objects and then inserts them at arbitrary positions in the target images.

3. The Proposed Framework

An overview of our framework is presented in Figure 1. Relying solely on the person and vehicle detection sub-module is not viable due to the significantly smaller number of samples for P0 and P2 objects compared to other categories. Note that P0 and P2 objects refer to Passenger 0 and Passenger 2, as clarified in Table 1. Therefore, we integrate the head detection sub-module alongside other modules specifically to enhance the mAP score for P0 and P2 classes. The detection results from object detection module are associated in the object association module. We leverage the existing tracking algorithm to harvest vehicle direction information in the post-processing for tracking module. Finally, combining vehicle direction information and the relative position of bounding boxes, we perform score modification on possible P0 or P2 objects in the score correction module (see Figure 1). Due to the serious imbalance of training data, we use Copy and Paste Augmentation [10] to increase the diversity of the dataset. All modules and techniques are detailed in the following sections.

3.1. Object Detection Module

Our object detection module has two object detection sub-modules, namely, the person and vehicle detection sub-module and the head detection sub-module. Particularly, the former sub-module is responsible for detecting persons and vehicles, which is trained with the original dataset's nine classes shown in Table 2. Meanwhile, the latter sub-module, which is trained with the re-labeling dataset [7], aims to mitigate the impact of data imbalance and object occlusion on the object detection performance.

Person and Vehicle Detection. We intensively experiment with different detection model architectures include YoloV7 [34], YoloV8 [16] and Co-DETR [38]. The 2024 AI City Challenge Track 5 dataset [35] contains nine classes: Motorbike, DHelmet, DNoHelmet, P1Helmet, P1NoHelmet, P2Helmet, P2NoHelmet, P0Helmet, P0NoHelmet. During the experiments, we used YoloV7-D6 [34], YoloV8 with variants X, X-P2, X-P6 [16] and Co-DETR with backbone Swin-L [38].

Head Detection. In this sub-module, we use EfficientDet D7 [29] and Co-DETR [38], with Co-DETR being recognized as the leading model in performance during the competition, validated on the COCO test-dev set [20]. For head detection, we integrate the Swin-L backbone into the Co-DETR architecture.

In order to improve the detection performance of these two sub-modules, we adapt the Test Time Augmentation (TTA) technique [26] in the inference phase of both object detection sub-modules, as illustrated in Figure 1. Furthermore, based on our previous work [7], we choose Weighted Box Fusion (WBF) [28] as our ensemble technique to combine multiple model's detection outputs to reduce the variance of prediction, while increasing the robustness and stability. Note that this WBF ensemble method is employed for both object detection sub-modules, as shown in Figure 1.

3.2. Object Association Module

Due to the fact that the object detection module produces a diverse range of classes such as Head, Motorbike, DHel-

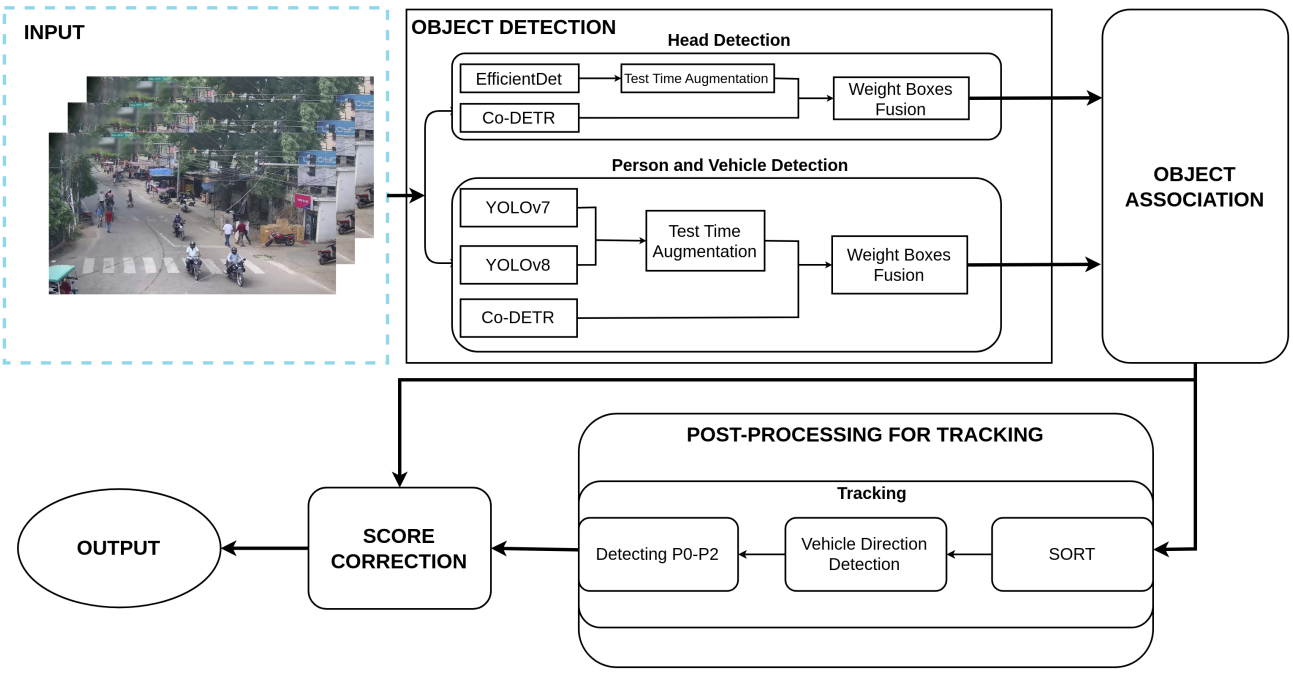


Figure 1. The proposed architecture for helmet violation detection in the 2024 AI City Challenge Track 5, which includes four different modules, as follows. First, the object detection module comprises two sub-modules for detecting motorcyclists' heads and helmets. Second, the object association module associates the outputs from these models to match their corresponding motorbikes with head and human objects. Third, the post-processing for tracking module tracks all the motorbikes to identify vehicle direction and find possible P0 and P2. Finally, the score correction module will adjust the confidence scores of P0 and P2. Here, P0 and P2 classes are defined in Table 1

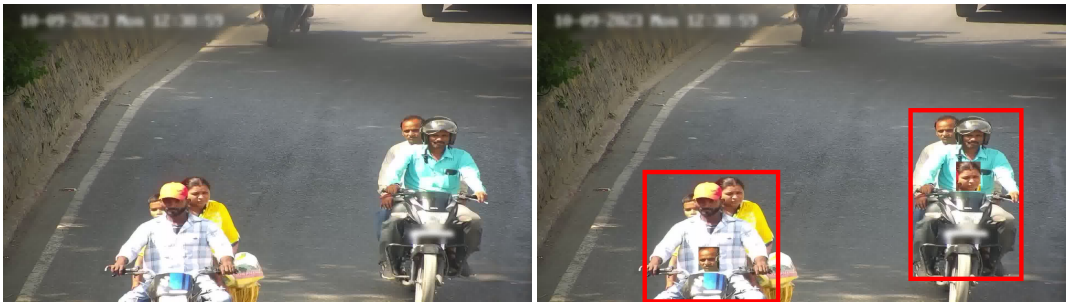
met, DNoHelmet, P1Helmet, P1NoHelmet, P2Helmet, P2NoHelmet, P0Helmet, and P0NoHelmet, we propose to use the object association module for assigning all possible pairs of human-motorbike and human-head similar to our previous work [7]. This association is achieved by analyzing the overlap areas and relative positions of bounding boxes with respect to the motorbikes. Consequently, the output comprises a list of motorbikes along with their belonging humans and heads. This output serves as input for the subsequent tracking module, which tries to track a group of vehicle, driver, heads, passengers by using the same tracking ID for them.

3.3. Post-processing for Tracking Module

Receiving outputs from the object association module, in this module, we propose to use a customized tracking module to capture vehicle direction information (i.e., motion towards or away from the camera's point of view). This information is then integrated with outputs from the head detection sub-module to potentially identify the presence of a P0 or P2 on the vehicle. Particularly, based on the SORT tracking method [3], we modify the Kalman Filter Estimation output similar to our previous work [7]. The identification of P0 or P2 instances within the video will serve as input data for the score correction module. Note that as presented in Section 3, we deliberately design this module in combination with the score correction module proposed in Section 3.4 to increase confidence scores of detected P0 and P2 objects, otherwise, these rare classes will severely degrade the overall detection performance.

Vehicle Direction Detection. We leverage a technique inspired by the work presented in [7] to extract vehicle direction information by analyzing the center point of the bounding box surrounding the motorbike in consecutive frames of the video feed. To determine the direction of the motorbike, we examine the displacement of its center point across frames. Specifically, if the motorbike is moving towards the camera, we set the direction flag to 1, indicating an inward direction (IN direction). By contrast, if the motorbike is moving away from the camera, the direction flag is set to 0, indicating an outward direction (OUT direction).

Detecting Possible P0 and P2 Objects. After detecting the vehicle direction, we develop a video-level classifier, which categorizes videos into two distinct classes: those that contain P2 or P0 and those that do not contain any of them. Herein, recall that P2 refers to passenger 2 (including P2Helmet and P2NoHelmet classes), while P0 refers to the child sitting in front of the driver (including P0Helmet and P0NoHelmet classes). The methodologies for detecting P2



(a) Original frame (b) Manual augmentation of adding P0 objects

(c) Original frame (d) Manual augmentation of adding P2 objects

Figure 2. Illustration for our manual augmentation method, where figures 2a and 2c show original frames, while figures 2b and 2d demonstrate samples generated by our manual augmentation of adding P0 and P2 objects.

and P0 objects are presented below.

- **Detecting Possible P2 Objects.** If the tracking ID in a video exhibits three bounding boxes of the head within 4 consecutive frames, subject to relative position constraints that the bounding boxes of the head placed at positions above the center of the motorbike's bounding box, the corresponding video will be labeled as containing P2.
- **Detecting Possible P0 Objects.** A video is categorized as containing P0 if, within 4 consecutive frames, a tracking ID demonstrates two head bounding boxes situated inside the driver's bounding box. Notably, these bounding boxes must adhere to relative position constraints, such that the bounding boxes of the head are placed close to the central vertical line of the driver's bounding box and above the center of the motorbike's bounding box. This categorization is further subject to the condition that the tracking ID is observed to move toward the camera.

3.4. Score Correction Module

Because the number of P0 and P2 objects is much less than that of the driver and P1 objects, the person and vehicle detection module tend to predict with very low confidence scores for these classes, leading to a low overall mAP score. Based on the classification results obtained from the previous modules, we implement a confidence score correction scheme for videos containing P0 or P2 objects, where an offset value is applied to the confidence scores of all P0 and P2 instances in these videos. Through experimentation on the validation set of Data V2 (defined in Section 4.1), the offset values for P0 and P2 instances are determined to be 0.1 and 0.2, respectively. Details about our proposed score correction module are shown in Algorithm 1.

Algorithm 1: Confidence Score Correction

Input : $person\_instances$: list of detected people in a video, each instance includes bounding box coordinates, object class ($.class$) and confidence score ($.conf$), $P0P2\_type\_videos$: list of videos containing P0 ($.hasP0$) or P2 ($.hasP2$) instances determined by the post-processing for tracking module.

Output: Updated $person\_instances$ confidence score which are P0 or P2 ($.conf$).

for *video in* $P0P2\_type\_videos$ **do**
  for *instance in* $video.person\_instances$ **do**
    if $instance.class$ *in* $[8, 9]$ *(see Table 1) and* $video.hasP0$ *is true* **then**
      $instance.conf \leftarrow instance.conf + 0.1$;
    if $instance.class$ *in* $[6, 7]$ *(see Table 1) and* $video.hasP2$ *is true* **then**
      $instance.conf \leftarrow instance.conf + 0.2$;
  end
end
return $person\_instances$;

3.5. Copy and Paste Augmentation Methods

Since P2NoHelmet and P0NoHelmet samples are very limited and only appear in a few videos, the model often tends to rely on the scene to predict these classes. As a result, this may cause overfitting problems, which degrade the detection performance over the validation and test dataset. We solve this problem by creating more frames with diverse backgrounds for these rare classes, utilizing the training data. This forces the model to learn the difference between two identical frames containing different objects. For this, we propose to use both manual and automatic copy and paste augmentation methods, as will be detailed next.

Manual Copy and Paste Augmentation. The manual augmentation method for adding both P2NoHelmet and P0NoHelmet objects is illustrated via Figure 2. We describe this method for enriching both P2NoHelmet and P0NoHelmet samples in detail in the following.

- **Enriching Data for P2NoHelmet.** Observing the training data, we found the fact that none of the training samples contained P2Helmet objects. Hence, it is essential to generate more data samples containing P2NoHelmet objects. To achieve this, we first select P1NoHelmet frames that have an IOU score with a driver greater than 0.5. This is due to the fact that P2 always sits behind P1. We then crop someone's head in the same selected frame and paste it in the space between the driver and P1NoHelmet. Note that the IOU score greater than 0.5 explains the lack of legs and arms of the new person. As a result, the P1NoHelmet object now becomes a new P2NoHelmet object. Note that the label of the added person is skipped because this person has a high overlap score with the driver and the new P2NoHelmet. Finally, we note that if there are many frames that come from the same video, only a few typical frames are selected for manual augmentation. Such manual augmentation for adding P2 objects is illustrated via Figure 2d.
- **Enriching Data for P0NoHelmet.** Similar to P2NoHelmet above, we try to generate additional training samples that contain P0NoHelmet objects. We first choose frames with NoHelmet objects and prioritize ones with P1Helmet or P1NoHelmet because P0NoHelmet often goes with one person behind the driver. Then, we take the NoHelmet head or head with shoulders and lower body, resize it accordingly, paste it in front of and lower than the driver's head, and draw a suitable bounding box. Observing the training set, we found that there are cases where children wear clothes of the same color as the driver. Thus, it is difficult to identify their arms and legs. Therefore, by adding additional heads of P0NoHelmet, we aim to instruct the model to focus on recognizing the head of this class. Such manual augmentation for adding P2 objects is illustrated via Figure 2b.

Automatic Copy and Paste Augmentation. Similar to the concept of Manual Copy and Paste Augmentation, we utilize a segmentation model [17] to grab one or more humans and paste them into suitable locations. The position of the added person relative to other people on the same vehicle determines whether the person's label is Passenger 0, Passenger 1 or Passenger 2, while their Helmet or NoHelmet label is retained. Different from the manual augmentation method, in this automatic method, we do not take into account the overlap constraint and always label the bounding box of the added persons. We provide two samples generated by the automatic augmentation technique in Figure 3.

4. Experiment Results

4.1. Dataset

Data Preparation: The object detection models in the person and vehicle detection sub-module are trained on two datasets, V1 and V2. Data V1 includes 2023 AI City Challenge Track 5 dataset [23] corrected erroneous labels by [7] along with our changes to P2 and P0 classes, addressing an absence of definition for the P0 class within the original 2023 dataset, and 2024 AI City Challenge Track 5 dataset [35]. Data V2 is based on Data V1 and filters out duplicate videos while further editing incorrect labels of P0 and P2 on the 2024 AI City Challenge Track 5 dataset [35] and adding samples generated by our copy and paste augmentation methods presented in Section 3.5. Details about the dataset are shown in Table 2. It is worth noting that the head bounding box is an important factor in the improvement of accuracy. Therefore, we reuse the head dataset from our previous work [7].

Table 2. Distributions of Data V1 and Data V2

ID	Class	Number of Instances			
		Data V1		Data V2	
		Train	Val	Train	Val
1	Motorbike	61238	9295	37168	6910
2	DHelmet	45002	6802	26619	5440
3	DNoHelmet	12842	2250	8346	830
4	P1Helmet	127	100	167	56
5	P1NoHelmet	8903	1139	5465	1265
6	P2Helmet	0	0	1	0
7	P2NoHelmet	138	10	194	46
8	P0Helmet	0	0	0	0
9	P0NoHelmet	146	47	169	68

Data Augmentation: We exclusively apply the copy and paste augmentation techniques to Data V2, where the numbers of generated samples for P2NoHelmet and P0NoHelmet classes are shown in Table 3 with 45 new objects for P2NoHelmet and 58 new objects for P0NoHelmet.

(a) Original frame (b) Automatic augmentation frame

(c) Original frame (d) Automatic augmentation frame

Figure 3. Illustration for our automatic augmentation method, where figures 3a and 3c demonstrate the original frames, and figures 3b and 3d show samples generated by our automatic augmentation of adding P0 and P2 objects.

Note that the augmented data samples are shown in Figure 2 for the manual augmentation and Figure 3 for the automatic augmentation method.

Table 3. Numbers of additional data samples generated by applying our copy and paste augmentation method to Data V2

Class ID	Class Name	Number of Instances	
		Original	Augmentation
7	P2NoHelmet	149	194 (+45)
9	P0NoHelmet	111	169 (+58)

4.2. Implementation Detail

4.2.1 Person and Vehicle Detection

We employ 3 architectures for person and vehicle detection, namely, YOLOv7 [34], YOLOv8 [16], and Co-DETR [38]. In particular, YOLOv7 is initialized with COCO pre-trained weights and subsequently fine-tuned using Data V1. The training process of YOLOv7 is conducted over 120 epochs utilizing 8 NVIDIA RTX 2080Ti 12GB GPUs. The image size is 1280 pixels, and the model size of YOLOv7 is D6. The initial learning rate is set at 1e-2, while OneCycleLR is employed as the decayed learning rate scheme. For consistency, the same image size is maintained during the inference phase.

Next, the YOLOv8 [16] model leverages pre-trained weights from OpenImage v7 [18] and undergoes fine-tuning with Data V1 and V2. We train two models of size X, employing a multi-scale image size approach with the initial learning rate at 1e-2 and the OneCycleLR learning rate scheme. The training procedure of YOLOv8 [16] is conducted across 8 NVIDIA RTX 2080Ti 12GB GPUs, spanning a training duration of 240 epochs. Moreover, two models of the size X-P6 and X-P2 are trained exclusively on Data V1, with an input size of 1280 pixels. After training, we select the optimal models for further evaluation, running inference on input size 832 pixels utilizing the TTA technique.

Finally, the Co-DETR model [38] is initialized with pre-trained weights from the Object 365 dataset [27] and then fine-tuned on Data V1 and V2, employing a learning rate of 1e-4. The training process of this model is carried out on 4 NVIDIA A100 40GB GPUs over 20 epochs. In the inference process, we apply an image scale of (2048, 1920).

4.2.2 Head Detection

For detecting heads, we employ 2 architectures: Efficient-Det D7 [29] having weights from [7] without fine-tuning and Co-DETR [38] with COCO pre-trained weights. Co-DETR [38] is fine-tuned specifically on head annotations extracted from [7], employing 20 epochs of training with a learning rate of 1e-4 across 4 NVIDIA A100 40GB GPUs. The scale of (2048, 1920) is applied in the inference process.

4.3. Evaluation Metrics

The evaluation metric for the 2024 AI City Challenge - Track 5 is the mean Average Precision (mAP@0.5) [35]. This metric quantifies the area under the Precision-Recall curve across all object classes. Note that this metric was introduced in the PASCAL VOC 2012 competition [8].

4.4. Object Detection Performance

Head Detection. The mAP performance of two head detection models are compared in Table 4, which shows that Co-DETR outperforms EfficientDet D7 by 6.6% mAP on the head validation set [7]. We expect to further improve the head detection sub-module accuracy by ensembling these two models using the WBF method [28], as illustrated in Figure 1 in Section 3.

Table 4. Performance of two head detection models on the head validation set

Model	mAP
EfficientDet D7 [7]	0.626
Co-DETR	**0.692**

Person and Vehicle Detection. The mAP performance of different person and vehicle detection models are demonstrated in Table 5. As expected, the Co-DETR model archives the highest score on the Data V1 validation set, surpassing other methods by a large margin.

Table 5. Performance of object detection models on the Data V1 validation set

Model	mAP
YOLOv8x-P6	0.5914
YOLOv8x-P2	0.5855
YOLOv8x	0.5784
YOLOv7-D6	0.6263
Co-DETR	**0.6786**

The Impact of Proposed Methods. In Table 6, we investigate the impact of different proposed methods, such as copy and paste augmentation, ensemble, and score correction, on the mAP performance. It is worth noting from this table that our proposed data augmentation facilitates a notable enhancement of 3.34% (mAP) score over the baseline, although adding a very small amount of samples for P2NoHelmet and P0NoHelmet (see Table 3). Furthermore, we selected 11 checkpoints from three distinct model architectures, YOLOv7, YOLOv8, and Co-DETR, to apply the WBF ensemble technique, using a non-maximum suppression threshold of 0.7 and a score threshold of 0.01. As seen via Table 6, our proposed ensemble solution further increases the performance by 2.08% mAP from 0.4383 to 0.4591. Finally, using the proposed score correction module, our final solution achieves a commendable score of 0.4792, which is 7.43% mAP higher than that of the Co-DETR baseline, as depicted in Table 6.

Table 6. Ablation study on the impact of proposed methods: copy and paste augmentation (CPA), ensemble, and P2 and P0 score correction, respectively. The baseline is the Co-DETR model.

CPA	Ensemble	Score correction	mAP
			0.4049 (baseline)
✓			0.4383 (+3.34%)
✓	✓		0.4591 (+2.08%)
✓	✓	✓	**0.4792** (+2.01%)

Comparison with Other Teams. Our proposed method has been submitted to the evaluation system, where Table 7 shows that our solution has achieved a final mAP score of 0.4792, ranking 3rd among participating teams of Track 5 in the AI City Challenge 2024.

Table 7. Leaderboard of Track 5 in the AI City Challenge 2024

Team ID	mAP
99	0.4860
76	0.4824
9 (Our)	**0.4792**
155	0.4675
5	0.4644

5. Conclusion

We proposed a novel deep learning framework for helmet violation detection, which combines different robust techniques, such as data augmentation, score correction, and ensemble methods, in order to effectively mitigate the impact of imbalanced data, significantly improving the detection performance over the Co-DETR baseline. As a result, our proposed approach achieves a remarkable 7.43% mAP improvement over the Co-DETR baseline and ranks 3rd on the private leaderboard with a final mAP score of 0.4792.

6. Acknowledgment

The work was sponsored by Vietnam Posts and Telecommunications Group (VNPT). We would like to thank Mr. Dien Hy Ngo, Deputy General Director of the Group, for his constant encouragement and support to the research team. We also express our gratitude to the AI Lab department of VNPT AI for providing the DGX A100 infrastructure for model training. Finally, we are thankful to Dr. Hung T. Le from VNPT AI for his assistance in reviewing this paper.

References

[1] Armstrong Aboah, Bin Wang, Ulas Bagci, and Yaw Adu-Gyamfi. Real-time multi-class helmet violation detection using few-shot data sampling technique and yolov8. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5349–5357, 2023. 2

[2] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face recognition with local binary patterns. In *Computer Vision-ECCV 2004: 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part I 8*, pages 469–481. Springer, 2004. 2

[3] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE international conference on image processing (ICIP)*, pages 3464–3468. IEEE, 2016. 1, 2, 4

[4] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 2

[5] Shun Cui, Tiantian Zhang, Hao Sun, Xuyang Zhou, Wenqing Yu, Aigong Zhen, Qihang Wu, and Zhongjiang He. An effective motorcycle helmet object detection framework for intelligent traffic safety. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5469–5475, 2023. 1, 2

[6] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05)*, pages 886–893. Ieee, 2005. 2

[7] Viet Hung Duong, Quang Huy Tran, Huu Si Phuc Nguyen, Duc Quyen Nguyen, and Tien Cuong Nguyen. Helmet rule violation detection for motorcyclists using a custom tracking framework and advanced object detection techniques. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5380–5389, 2023. 1, 2, 3, 4, 6, 7, 8

[8] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 8

[9] Ruben J Franklin et al. Traffic signal violation detection using artificial intelligence and deep learning. In *2020 5th international conference on communication and electronics systems (ICCES)*, pages 839–844. IEEE, 2020. 1

[10] Golnaz Ghiasi, Yin Cui, Aravind Srinivas, Rui Qian, Tsung-Yi Lin, Ekin D Cubuk, Quoc V Le, and Barret Zoph. Simple copy-paste is a strong data augmentation method for instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2918–2928, 2021. 3

[11] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015. 2

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014. 3

[13] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017. 2

[14] Marti A. Hearst, Susan T Dumais, Edgar Osuna, John Platt, and Bernhard Scholkopf. Support vector machines. *IEEE Intelligent Systems and their applications*, 13(4):18–28, 1998. 2

[15] Dan Hendrycks, Norman Mu, Ekin D Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. *arXiv preprint arXiv:1912.02781*, 2019. 3

[16] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, 2023. 2, 3, 7

[17] Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4015–4026, 2023. 1, 6

[18] Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*, 2020. 7

[19] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs]*, 2015. arXiv: 1504.01942. 2

[20] Tsung-Yi Lin, Michael Maire, Serge J. Belongie, Lubomir D. Bourdev, Ross B. Girshick, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. 2, 3

[21] Wei Liu, 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*, pages 21–37. Springer, 2016. 1, 2

[22] Vishal Mandal, Abdul Rashid Mussah, Peng Jin, and Yaw Adu-Gyamfi. Artificial intelligence-enabled traffic monitoring system. *Sustainability*, 12(21):9177, 2020. 1

[23] Milind Naphade, Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Qi Feng, Vitaly Ablavsky, Stan Sclaroff, Pranamesh Chakraborty, Sanjita Prajapati, Alice Li, Shangru Li, Krishna Kunadharaju, Shenxin Jiang, and Rama Chellappa. The 7th ai city challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2023. 2, 6

[24] Leif E Peterson. K-nearest neighbor. *Scholarpedia*, 4(2): 1883, 2009. 2

[25] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object de-

tection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 1, 2

[26] Divya Shanmugam, Davis Blalock, Guha Balakrishnan, and John Guttag. Better aggregation in test-time augmentation, 2021. 3

[27] Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8430–8439, 2019. 7

[28] Roman A. Solovyev and Weimin Wang. Weighted boxes fusion: ensembling boxes for object detection models. *CoRR*, abs/1910.13302, 2019. 2, 3, 8

[29] Mingxing Tan, Ruoming Pang, and Quoc V Le. Efficientdet: Scalable and efficient object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10781–10790, 2020. 2, 3, 7

[30] Duong Nguyen-Ngoc Tran, Long Hoang Pham, Hyung-Joon Jeon, Huy-Hung Nguyen, Hyung-Min Jeon, Tai Huu-Phuong Tran, and Jae Wook Jeon. Robust automatic motorcycle helmet violation detection for an intelligent transportation system. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5341–5349, 2023. 1, 2

[31] Chun-Ming Tsai, Jun-Wei Hsieh, Ming-Ching Chang, Guan-Lin He, Ping-Yang Chen, Wei-Tsung Chang, and Yi-Kuan Hsieh. Video analytics for detecting motorcyclist helmet rule violations. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5365–5373, 2023. 2

[32] Hoang Van Truong. Motorbike generator using bidirectional generative adversarial networks. In *Proceedings of the 2020 International Conference on Computer Communication and Information Systems*, pages 40–44, 2020. 3

[33] Bor-Shiun Wang, Ping-Yang Chen, Yi-Kuan Hsieh, Jun-Wei Hsieh, Ming-Ching Chang, JiaXin He, Shin-You Teng, HaoYuan Yue, and Yu-Chee Tseng. Prb-fpn+: Video analytics for enforcing motorcycle helmet laws. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 5476–5484, 2023. 2

[34] Chien-Yao Wang, Alexey Bochkovskiy, and Hong-Yuan Mark Liao. Yolov7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7464–7475, 2023. 2, 3, 7

[35] Shuo Wang, David C. Anastasiu, Zheng Tang, Ming-Ching Chang, Yue Yao, Liang Zheng, Mohammed Shaiqur Rahman, Meenakshi S. Arya, Anuj Sharma, Pranamesh Chakraborty, Sanjita Prajapati, Quan Kong, Norimasa Kobori, Munkhjargal Gochoo, Munkh-Erdene Otgonbold, Ganzorig Batnasan, Fady Alnajjar, Ping-Yang Chen, Jun-Wei Hsieh, Xunlei Wu, Sameer Satish Pusegaonkar, Yizhou Wang, Sujit Biswas, and Rama Chellappa. The 8th AI City Challenge. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, 2024. 1, 3, 6, 8

[36] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017. 1

[37] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6023–6032, 2019. 3

[38] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6748–6758, 2023. 2, 3, 7