A Coarse-to-fine Two-stage Helmet Detection Method for Motorcyclists

Hongpu Zhang[1], Zhe Cui[1,2*], Fei Su[1,2]
[1]Beijing University of Posts and Telecommunications
[2]Beijing Key Laboratory of Network System and Network Culture, China
{zhp,cuizhe,sufei}@bupt.edu.cn

Abstract

In recent years, motorcycle accidents have occurred frequently, with a important reason being that motorcyclists do not wear helmets properly. The visual method of detecting whether a motorcyclist is wearing helmet based on monitoring videos can provide technical support for traffic management. However, the appearance characteristics of motorcycle drivers and passengers are too similar to distinguish, which makes it difficult to detect helmet. In this task, we propose ***a Coarse-to-fine Two-stage Helmet Detection Method for Motorcyclists*** *to improve the accuracy of helmet and motorcyclist detection. Our Coarse detector detect the rough location of people and motorcycle as the initial suggestion for the following Fine-grained detection. Then our Fine-grained detector employs a classification branch to accurately distinguish between the driver and passengers. Finally, we use some useful strategies such as Test Time Augmentation (TTA) and Weighted Boxes Fusion (WBF) to achieve further improvements to our proposed framework. Our proposed framework achieved mAP score of 39.4 % on the test dataset of AI City Challenge 2024 Track5.*

1. Introduction

Motorcycles are one of the most popular modes of transportation, particularly in developing countries such as India. Due to lesser protection compared to cars and other standard vehicles, motorcycle riders are exposed to a greater risk of crashes. Therefore, wearing helmets for motorcycle riders is mandatory as per traffic rules and automatic detection of motorcyclists without helmets is one of the critical tasks to enforce strict regulatory traffic safety measures [16]. Detecting whether motorcyclists are wearing helmets automatically based on monitoring videos is a valuable visual research task. The Track 5 of 2024 AI City Challenge [28], Detecting Violation of Helmet Rule for Motorcyclists, provides an evaluation platform for this research. This task requires detecting motorcyclists with or without helmets and each rider in a motorcycle (driver, passenger1, passenger2, passenger0) should be separately identified if they have a helmet or not. There are a total of 9 categories: motorbike, DHelmet, DNoHelmet, P1Helmet, P1NoHelmet, P2Helmet, P2NoHelmet, P0Helmet, P0NoHelmet. The evaluation metric is mAP across all frames in the videos.

The training and testing datasets provided by the organizers are both traffic monitoring videos. As shown in Fig. 1, for this object detection task, the main issues with these videos are: (1) low video quality. Due to blurry frames and noise caused by limitations of monitoring devices and the environment; (2) varying object sizes. Owing to the fixed position and large field of view of the monitoring cameras, objects in the distance appear smaller, making it difficult to detect and determine whether a helmet is being worn; (3) complex traffic conditions, such as high vehicle density, which can result in occlusions and intersections, making it more difficult to detect objects.

Currently, existing motorcycle detection methods [1, 6, 8, 24–26] followed the typical approach of object detection and multiple object tracking, which consists of several components. The first component is object detection, and most studies used an ensemble model to improve the performance and generalization. Then, object association or identification was used to correctly locate the driver/passengers. Some previous methods determined the location passenger2 based on tracking, but now it is also necessary to detect passenger0, so tracking cannot be used to determine passenger2. Finally, Category Refine modules were used to generate the results and correct any misclassified classes. Nevertheless, the aforementioned methods fall short in identifying passenger0.

Thus, we propose **a Coarse-to-fine Two-stage Helmet Detection Method for Motorcyclists**. To obtain a high recall object detection model on the dataset, we treat people and motorcycles that ride together as a whole object in the Coarse detector, which has a larger detectable box and can be considered as a single object detection category

*Corresponding author

(a) low video quality. (b) varying object sizes. (c) complex traffic conditions.

Figure 1. Problems exit in the training data videos.

during training and predicting. This approach can alleviate the problem of detecting small objects. Subsequently, in order to better distinguish driver and passenger with similar appearance characteristics, we add an additional classification branch in the Fine-grained detector. Finally, we post-process the results to reduce false positives. Furthermore, we also apply some strategies to improve detection performance, including Test Time Augmentation (TTA) and Weighted Boxes Fusion (WBF) [23].

In summary, the main contributions of this study are as follows:

- We propose **a Coarse-to-fine Two-stage Helmet Detection Method for Motorcyclists** to detect motorcycles and each rider in a motorcycle (driver, passenger1, passenger2, passenger0) should be separately identified if they have a helmet or not.
- We add an additional classification branch to Co-DETR (the Fine-grained detector) to enhance its capability in predicting confusing categories. Additionally, we provide a useful package of tips for object detection task, including Test Time Augmentation (TTA) and Weighted Boxes Fusion (WBF).
- On the AI City Challenge Track5 test dataset, our proposed framework achieves 39.4 % (mAP), which ranks 7th in the 2024 AI City Challenge Track 5.

2. Related Work

2.1. Object Detection

Object detection is an important task in computer vision, which aims to recognize and locate specific objects from images or videos. In recent years, many excellent object detection algorithms have been proposed, including SSD [14], Fast RCNN [9], Faster RCNN [21], YOLO [3, 11–13, 18–20, 27], DETR [4], Co-DETR [31].

Fast RCNN [9], proposed by Girshick et al. in 2015, introduced end-to-end detector training on shared convolutional features, achieving compelling accuracy and speed. Faster RCNN [21], proposed by Ren et al. in the same year, improved upon Fast RCNN by introducing a two-stage detection model with a Region Proposal Network (RPN). SSD [14] is a one-stage object detection algorithm proposed by Liu et al. in 2016 that employs convolutional neural network for simultaneous detection and position regression. YOLO [18] is proposed by Joseph in 2016. Unlike traditional object detection algorithms, YOLO achieves object detection with a single forward propagation, enabling end-to-end training and real-time speeds while maintaining high precision. This feature has made YOLO widely popular in practical applications. However, due to the coarse output of the network, it performs poorly in detecting small and dense objects. After that, people continued to improve it and gradually derived subsequent versions, and the detection performance improve steadily. YOLOv8 [12] adopts a more gradient-rich C2f structure on the basis of YOLOv5 [11], and replaces the head with the current mainstream decoupling head structure, which separates the classification and detection heads. These improvements have effectively enhanced accuracy. Therefore, we adopt YOLOv8 as our Coarse detector based on its outstanding performance.

In addition to the above methods based on convolutional networks, the transformer architecture has been recently applied for object detection as backbone, which achieves a remarkable balance between speed and accuracy, such as Vision Transformer(ViT) [7] and Swin Transformer [15]. Detecting Objects with Transformers (DETR) [4] is the first transformer-based detector proposed in 2020, which surpassed state-of-the-art on the COCO dataset. Since then, several variants and alternatives has been proposed, such as Co-DETR [31], a collaborative hybrid assignment training scheme. The key insight of Co-DETR is to use versatile one-to-many label assignments to improve the training efficiency and effectiveness of both the encoder and decoder. This strategy can effectively improve the mAP. Thus, we adopt Co-DETR as our Fine-grained detector.

2.2. Motorcyclist Helmet Detection

Helmet violation detection, also known as helmet enforcement, has become an important area of research in recent years due to the importance of helmet usage in reduc-

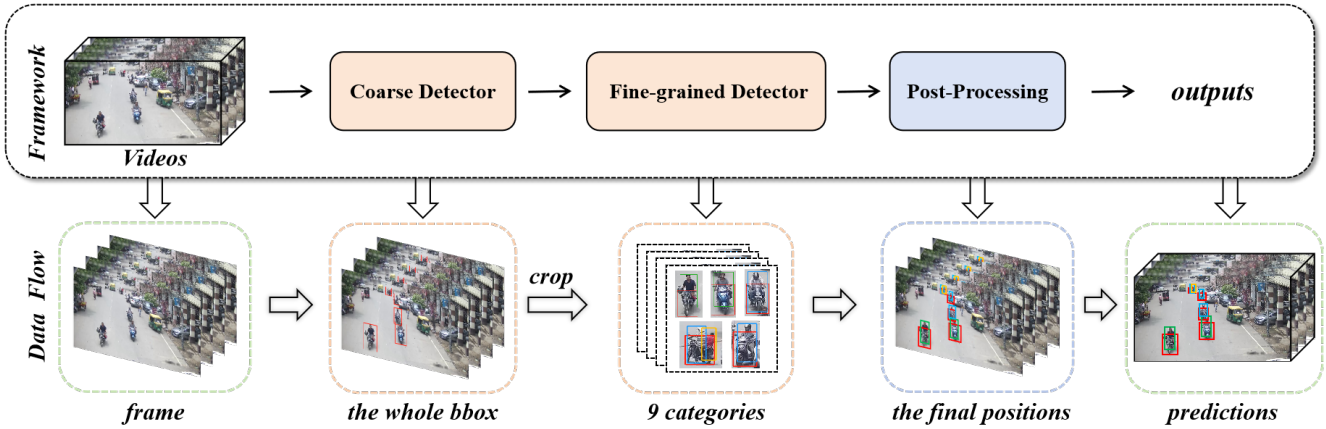


Figure 2. The pipeline of the framework. The input of this framework is video frames. First, **the Coarse detector** is utilized to detect the whole bounding box of people and motorcycles. Subsequently, we crop the bounding box and send them to **the Fine-grained detector** to identify the individual components of the object, including the motorbike, driver, and passenger, and whether or not they were wearing a helmet. Finally, we perform WBF to obtain the final prediction result.

ing injury and fatalities in road accidents. In the 2023 AI City Challenge Task 5, many researchers have done a lot of research on it [1, 6, 8, 24–26].

A study [6] introduced a robust Motorcycle Helmet Object Detection (MHOD) framework with model ensemble which achieved first place. Their approach involved utilizing a model ensemble based on the DETA algorithm to enhance performance and generation. Subsequently, they conducted the Passenger Recall Module (PRM) to improve the recall of the passenger category. Finally, they employed the Category Refine Module (CRM) to generate results and rectify any misclassified classes. Another research [24] focused on utilizing the detector based YOLOv8 to focus on first localizing the entire motorbike with a person on it, and then identifying the individual components of the object, including the motorbike, driver, and passenger, and whether or not they were wearing a helmet. In a separate study, a system [8] was proposed to process video streams frame-by-frame through three components, including Object detection, Object association, and Post-processing for tracking module. The object detection component is responsible for detecting all necessary objects in each frame, while the object association component connects each driver/passenger to the corresponding motorcycle and identifies the number of humans on the motorcycle. Finally, they design a post-processing for tracking approach to utilize object information to accurately reassign human classes, resulting in significant improvement in overall system performance.

2.3. Object Tracking

Object tracking is an important research direction in the field of computer vision, which aims at automatically tracking the motion trajectory of the objects in video sequences. Currently, various methods including FairMOT [30] based on one-shot MOT, TransTrack [5] which utilizes attention mechanism, as well as SORT [2] and DeepSORT [29] which uses tracking-by-detection have been widely employed.

FairMOT [30] is a joint detection and tracking algorithm that utilizes the Siamese network for feature extraction and similarity measurement for object matching. However, it has high computational complexity. TransTrack [5] is an attention-based tracking algorithm using Transformer encoders and decoders, capturing motion and semantic relationships but requiring more resources. SORT [2] is a lightweight tracking algorithm using Kalman filters and the Hungarian algorithm for object tracking. DeepSORT [29] introduces deep learning models on the basis of SORT, mainly to address problems such as object overlap and occlusion. DeepSORT uses a convolutional neural network for feature extraction of objects and a cosine similarity for tracking object matching. DeepSORT can track objects more accurately and improve object re-identification accuracy. To better balance tracking efficiency and accuracy, we selected DeepSORT as the tracker.

3. Method

3.1. Overall Architecture

The framework of the Coarse-to-fine two-stage helmet fetection method for motorcyclists proposed in this paper is illustrated in Fig. 2. Due to the varying object sizes in traffic monitoring videos, where objects appear smaller in the distance and larger nearby, detecting the whole image at once will miss the targets and lower the accuracy. Therefore, we propose to divide this task into two stages. In the first stage,

the Coarse detector is utilized to detect the whole bounding box of people and motorcycles. Subsequently, in the second stage, we crop the bounding box and send them to **the Fine-grained detector** to identify the individual components of the object, including the motorbike, driver, and passengers, and whether or not they were wearing a helmet. Finally, we propose some post-processing strategies to filter out the false positives, including the score fusion, tracking correction, and model fusion. The results are combined based on the confidence scores from these two stages to retain the correct targets. Additionally, we utilize object tracking to correct false positives. And by fusing the predictions of multiple detectors, the accuracy improves. The design of each stage will be detailed in the following sections.

3.2. Coarse Detector

In order to eliminate the interference of background, we utilize a Coarse detector in the first stage to firstly identify the bounding box of people and motorcycles. In view of the higher accuracy and faster inference speed of the YOLOv8, we adopt the YOLOv8x, which is the largest version, as our motorcycle ROI detector.

The original AI City Challenge Track 5 dataset contains ground truth for 9 categories, whereas our coarse detector dataset only requires ground truth for the overall bounding boxes of people and motorbike. Therefore, in this stage, we regenerate the dataset by matching motorcycles with the people riding on them to create a new large bounding box. As shown in the Fig. 3, by observing the ground truth of the original dataset, we find that regardless of the direction in which the motorbike is traveling, the bottom edge of the person's bounding box always lies above the bottom edge of the motorcycle's bounding box. Based on this constraint, we further compute the IoU (Intersection over Union) between motorcycles and people, considering them as the matching pairs when the IoU exceeds a specific threshold. The results of these matching pairs are shown in Fig. 5 (b). Then we send the regenerated dataset to the coarse detector for training.

After detecting the required ROI regions from the entire image, we get not only the cropped images of the whole bounding box containing the people and the motorbikes they ride but also the corresponding coordinates on the input image.

$$D_t = \{d_{t,1}, d_{t,2}, \ldots, d_{t,n}\} \tag{1}$$

where D_t denotes all bounding boxes detected in an image at time t and $d_{t,i}$ is one of them.

$$d_{t,i} = \left\{x^D_{t,i}, y^D_{t,i}, w^D_{t,i}, h^D_{t,i}, s^D_{t,i}\right\} \tag{2}$$

where $(x^D_{t,i}, y^D_{t,i})$ represents the center point, $(w^D_{t,i}, h^D_{t,i})$ denotes the width and height, and $s^D_{t,i}$ is the confidence score of the bounding box at index i in time t.

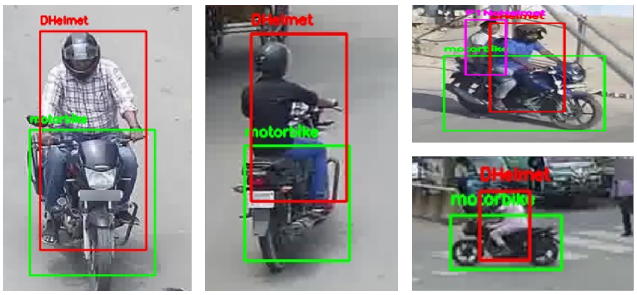


Figure 3. The ground truth of the original dataset. The bottom edge of the person's bounding box always lies above the bottom edge of the motorcycle's bounding box.

The above information is used for the Fine-grained detector and post-processing.

3.3. Fine-grained Detector

Upon obtaining the overall bounding box of people and motorbikes from the Coarse detector, we employ Co-DETR as the Fine-grained detector to detect motorcycle, driver, passenger, and whether or not they were wearing a helmet. Due to the collaborative hybrid assignment training scheme of Co-DETR, it can easily enhance the learning ability of both the encoder and decoder.

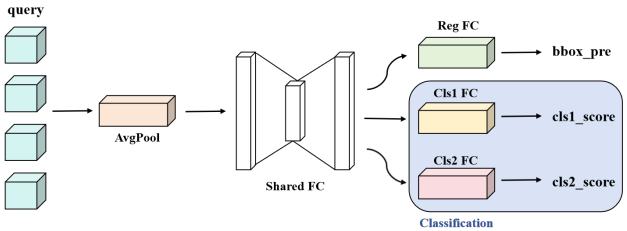


Figure 4. The head architecture. We employ two classification branch in the Fine-grained detector. One branch is employed for classifying motorbike, driver or passenger, the other branch is dedicated to determining whether they are wearing a helmet.

In this detection task, it can be effectively divided into two distinct tasks: distinguishing whether a motorcyclist is a driver or a passenger, and discerning whether they are wearing a helmet. The differences can significantly affect network performance. Therefore, we exploit two separate classification branch to handle these tasks, as illustrated in Fig. 4. One branch is employed for classifying driver or passenger, the other branch is dedicated to determining whether they are wearing a helmet. Adding an extra classification branch mainly helps to extract better feature representation of the shared layer for similar classes distinguishing. The total classification loss is as follows.

$$L_{cls} = L_{cls_1} + L_{cls_2} \tag{3}$$

(a) The original dataset of ground truth. (b) The Coarse detector dataset. (c) The Fine-grained detector dataset.

Figure 5. The conversion of the dataset.

where L_{cls_1} and L_{cls_2} represent the two classification branch losses. We use Binary Cross Entropy (BCE) to supervise the classification. Through the above two independent classification branches, we can better classify similar categories and improve network model performance.

After processing the overall bounding boxes obtained from the Coarse detector through the Fine-grained detector, we get the result of 9 categories:

$$C_{t,i} = \{c_{t,i,1}, c_{t,i,2}, \ldots, c_{t,i,m}\} \tag{4}$$

where $C_{t,i}$ denotes the detection results of cropped bounding box $d_{t,i}$.

$$c_{t,i,j} = \left\{x^C_{t,i,j}, y^C_{t,i,j}, w^C_{t,i,j}, h^C_{t,i,j}, s^C_{t,i,j}, c^C_{t,i,j}\right\} \tag{5}$$

where $(x^C_{t,i,j}, y^C_{t,i,j})$ represents the center point, $(w^C_{t,i,j}, h^C_{t,i,j})$ denotes the width and height, $s^C_{t,i,j}$ is the confidence score of the bounding box, and $c^C_{t,i,j}$ is the predicted category at index i, j in time t. Following the complete Fine-grained detector, we get the predictions C_t of each object in cropped images D_t.

$$C_t = C_{t,1} \cup C_{t,2} \cup \ldots \cup C_{t,n} \tag{6}$$

3.4. Post-Processing

In order to filter out the false positives, we propose some post-processing strategies, which consist of Score fusion, Tracking correction, and Model fusion. The Score fusion combines the results based on confidence scores to ensure the retention of accurate targets. The Tracking correction involves the utilization of object tracking to rectify false positives. The Model fusion (WBF) fuses bounding boxes generated by multiple detectors through weighted fusion to enhance performance. Each strategy will be further elaborated in the following sections.

3.4.1 Score Fusion

The results are combined based on the confidence scores from the Coarse detector and Fine-grained detector to retain the correct targets. Initially, we discard bounding boxes obtained from the Coarse detector with scores $s^D_{t,i}$ lower than the threshold (set to 0.2). Subsequently, for the qualifying predictions, we determine the final position and score of the predictions by employing C_t and D_t, where the score $s_{t,i,j}$ of each prediction is as follows (λ is set to 0.5).

$$s_{t,i,j} = \lambda \cdot s^D_{t,i} + (1 - \lambda) \cdot s^C_{t,i,j} \tag{7}$$

This approach helps to eliminate false positives and retain relatively accurate results.

3.4.2 Tracking Correction

Due to the fixed-camera surveillance videos in our dataset, objects appear larger when closer and smaller when father away, and occlusions are common, which will result in variations in the predicted category of an object in certain frame. Thus, we employ object tracking for correction to reduce the number of false positives.

Initially, we apply DeepSORT to associate predictions from Score fusion, obtaining the motion trajectories of people and motorcycles. Next, we calculate the frequency of predicted categories with the same track ID across all frames. When the frequency of a certain category c exceeds 50 % of the total detections, we consider it to be category c, and correct it to category c across all frames.

3.4.3 Model Fusion

Different detectors perform differently on different objects. We can use model fusion to merge the results of different detectors to improve performance. In our approach, we adopt WBF to fuse models. Firstly, we filter out the bounding boxes generated by each detector, removing those with confidence scores below a threshold. Secondly, we calculate the overlap between the remaining bounding boxes. Next, based on the confidence scores and overlap, we compute the weight for each bounding box. Finally, we perform weighted fusion on all bounding boxes according to their weights to generate the final detection results.

4. Experiment

4.1. Datasets

The training and testing datasets consist of 100 surveillance videos. Each video is 20 seconds duration, recorded at 10 fps. The video resolution is 1920×1080. This task requires detecting motorcyclists with or without helmets and each rider in a motorcycle should be separately identified if they have a helmet or not. There are 9 categories totally: motorbike, DHelmet, DNoHelmet, P1Helmet, P1NoHelmet, P2Helmet, P2NoHelmet, P0Helmet, P0NoHelmet. The evaluation metric is the mAP of all frames in the videos.

To better adapt to our proposed two-stage motorcycle detection, we convert the original AI City Challenge Track 5 dataset into two distinct datasets, each with different labels and image dimensions.

For **the Coarse detector**, we merge the 9 original classes from the dataset into a single class, covering the overall bounding boxes of people and motorbike they ride. For **the Fine-grained detector**, we crop image containing the overall bounding boxes of people and motorbike as training samples, while retaining the original 9 classes as labels. As shown in Fig. 5, (a) represents the ground truth of original dataset with 9 classes and 1920×1080 resolution; (b) is the dataset of transforming the initial dataset into the Coarse detector, with only 1 category of the overall bounding box of people and motorbike; (c) shows the cropped image of overall bounding box from (b), now with 9 classes.

4.2. Implementation Details

4.2.1 Training Phase

For the Coarse detector, we train our model at a resolution of 1280 using the largest version of YOLOv8 (YOLOv8x). Our model uses SGD as the optimizer with a default weight decay of 0.0005 and momentum of 0.937.

For the Fine-grained detector, we use Co-DETR with Swin-L backbone, which loads the pre-trained model parameters on Objects365 [22]. Learning rate is 1e-4 and weight decay is 1e-4. Intermediate size of the feedforward layers in the transformer blocks is 2048 and number of attention heads inside the transformer's attentions is 8. These parameters are default and we have not modified them. Since we are working exclusively with cropped images derived from the bounding box results of the Coarse detector, the training resolutions is resized to 640 instead of 1280.

4.2.2 Testing Phase

Details. In the testing phase, we extract each frame from the video of the test set to ensure the quality of the image. For the first stage, we employ the YOLOv8 model, which has been trained for 36 epochs. The confidence threshold is set to a low value of 0.1, which retains a large number of bounding boxes for subsequent processing in the next stage. And the inference resolution is 1280. For the second stage, we apply the Co-DETR model, which has been trained for 12 epochs. The whole bounding box of people and motorcycles obtained from the first stage inference is cropped and fed into this stage for inference, with a resolution of 640.

Test Time Augmentation. During the inference phase, we use Test Time Augmentation (TTA), which is an application of data augmentation to the test dataset, to improve the performance of our method. We first scale the image to 1280, and then test with images of different scales, where we downscale the scaled image to 1x, 0.83x, 0.75x, 0.67x, and 0.5x using 5 different scales. We randomly flipped images down to 0.83x and 0.75x. Finally, we feed 5 images of different scales to the Coarse detector, and use NMS to fuse the test predictions.

4.3. Evaluation Metric

The evaluation metric is based on mean Average Precision (mAP) across all frames in the test videos, as defined in the PASCAL VOC 2012 competition [10]. The mAP score computes the mean of average precision (the area under the Precision-Recall curve) across all the object classes.Bounding boxes with a height or width of less than 40 pixels and those that overlapped with any redacted regions in the frame were filtered out to avoid false penalization [17].

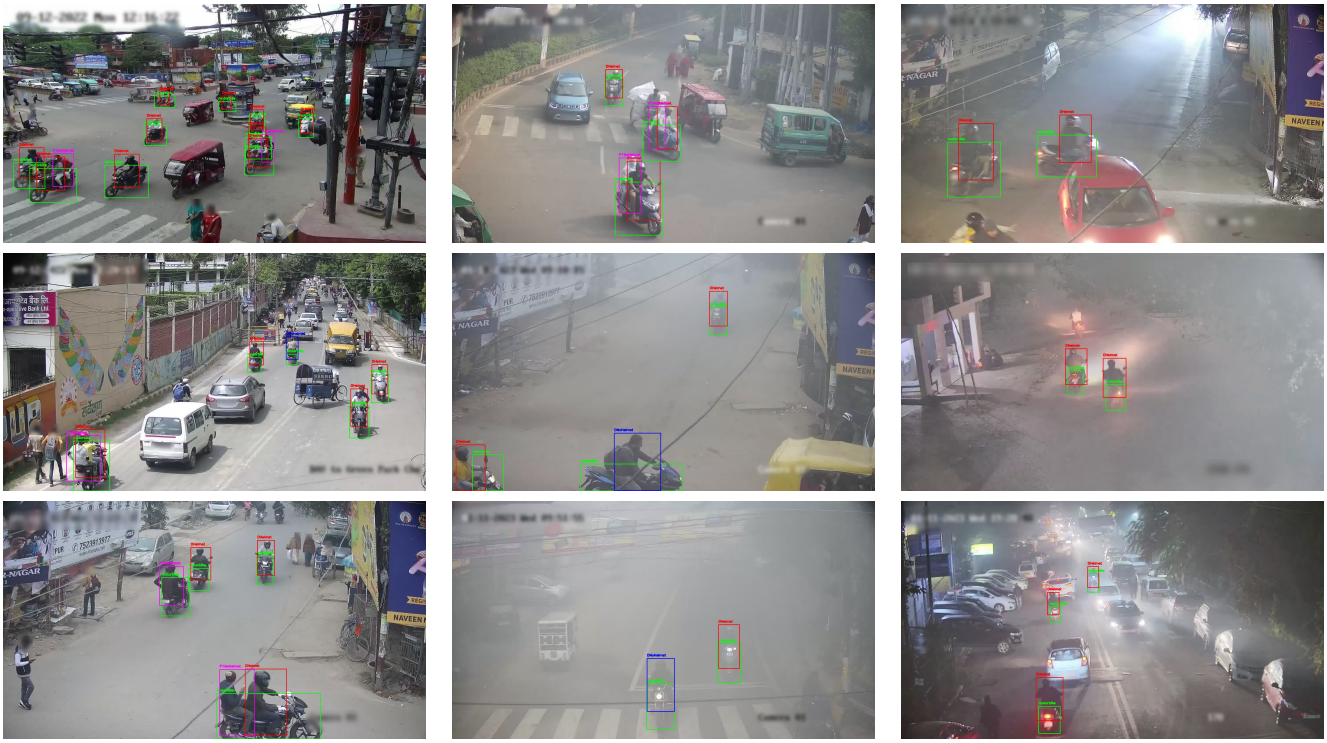
4.4. Experimental Results

In this section, we evaluate 2024 AI City Challenge Task 5 on the test set and compare with other methods. Table 1 reports our experimental results. The first row shows the result of using YOLOv8 as the Fine-grained detector, and the second row is choosing Co-DETR as the Fine-grained detector. We found that the performance on Co-DETR is better than YOLOv8, so we choose Co-DETR as our Fine-grained detector. The last row shows our final result. And the visualization of the final result is shown in Fig. 6.

Table 2 presents the final ranking result compare with other teams. The proposed system is submitted to the Track5 of 2024 AI City Challenge for evaluation. In the end, we get the 7th place with 39.4 % mAP.

Method	mAP
YOLOv8	22.4
Co-DETR	31.9
Ours	39.4

Table 1. Ablation study of different detectors in the second stage.



(a) daytime (b) haze (c) nighttime

Figure 6. The visualization of the result, including daytime, haze, and nighttime.

Rank	Team ID	Team Name	mAP
1	99	Helios	48.60
2	76	CMSR_PANDA	48.24
3	9	VNPT AI	47.92
4	155	TeleAI	46.75
5	5	SKKUAutoLab	46.44
6	228	DIDANO1	46.21
7	**57**	**BUPT_MCPRL**	**39.40**
8	247	CHTTL_IOTLAB	36.50
9	154	aio_tts	35.47
10	90	Graph@FIT+Comenius	34.65

Table 2. Top 10 Leaderboard of Track5 in the 2024 AI City Challenge.

Method	mAP
baseline	31.9
+Classification branch	33.4(+1.5)
+Model fusion	39.4(+6.0)

Table 3. Ablation study of each strategy.

4.5. Ablation Study

In order to verify the effectiveness of the proposed framework, we conduct ablation experiments on the test set of 2024 AI City Challenge Task 5. As shown in Table 3, we incrementally add modules from each layer to the baseline to demonstrate the contribution of these modules for improving model performance. The first row to the last row, mAP gradually increases from 31.9 % to 39.4 %.

Effect of Extra Classification Branch. Adding an extra classification branch to the original structure, which divides the classification into two groups, will make it easier to distinguish confusing categories. After introducing the classification branch, the mAP of the network has increased by 1.5 % compared with the original model, and the detection performance has been greatly improved.

Effect of Multi-model Fusion. Model fusion can integrate the learning capabilities of each model, so that the final result can complement each other and improve the generalization ability of the final model. In this challenge we employ Weighted Boxes Fusion (WBF) to combine the predictions of object detection models. We used multiple training models, including YOLOv8x, Co-DETR, Co-DETR with added classification branch. As shown in Table 3, the fused model has better performance on the test set.

5. Conclusion

The traffic monitoring videos have the characteristics of low video quality and small target in the distance, which brings challenges to general object detection. In response to these problems, we propose a novel Coarse-to-fine two-stage framework to detect motorcycle helmets, which eliminate background interference. In the first stage, namely **the Coarse detector**, the initial bounding box of person and motorcycles is detected. In the second stage, which is called **the Fine-grained detector**, we perform more detailed detection with an extra classification branch, which can distinguish easily confused categories. In addition, we have implemented some experienced tricks to better improve performance. Experimental results on the public test set of 2024 AI City Challenge Track5 demonstrate the effectiveness of our method, which achieves score of 39.4%, ranking 7th on the leaderboard.

6. Acknowledgement

This work is supported by Chinese National Natural Science Foundation under Grants 62076033.

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5349–5357, 2023. 1, 3

[2] 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. 3

[3] Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. *arXiv preprint arXiv:2004.10934*, 2020. 2

[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] Peng Chu, Jiang Wang, Quanzeng You, Haibin Ling, and Zicheng Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4870–4880, 2023. 3

[6] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5469–5475, 2023. 1, 3

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020. 2

[8] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5380–5389, 2023. 1, 3

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

[10] K He. The pascal visual object classes challenge 2012 (voc2012) results. 6

[11] Glenn Jocher. ultralytics/yolov5: v3.1 - Bug Fixes and Performance Improvements. https://github.com/ultralytics/yolov5, Oct. 2020. 2

[12] Glenn Jocher, Ayush Chaurasia, and Jing Qiu. Ultralytics YOLO, Jan. 2023. 2

[13] Chuyi Li, Lulu Li, Hongliang Jiang, Kaiheng Weng, Yifei Geng, Liang Li, Zaidan Ke, Qingyuan Li, Meng Cheng, Weiqiang Nie, et al. Yolov6: A single-stage object detection framework for industrial applications. *arXiv preprint arXiv:2209.02976*, 2022. 2

[14] 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. 2

[15] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10012–10022, 2021. 2

[16] Rame Chellappa Milind Naphade et al. aicity, 2024. https://www.aicitychallenge.org/. 1

[17] 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*, June 2023. 6

[18] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016. 2

[19] Joseph Redmon and Ali Farhadi. Yolo9000: better, faster, stronger. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7263–7271, 2017. 2

[20] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 2

[21] Shaoqing Ren, 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. 2

[22] 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. 6

[23] Roman Solovyev, Weimin Wang, and Tatiana Gabruseva. Weighted boxes fusion: Ensembling boxes from different object detection models. *Image and Vision Computing*, 107:104117, 2021. 2

[24] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5341–5349, 2023. 1, 3

[25] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5365–5373, 2023. 1, 3

[26] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5476–5484, 2023. 1, 3

[27] 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

[28] 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*, June 2024. 1

[29] 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. 3

[30] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129:3069–3087, 2021. 3

[31] 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