Computers, Materials & Continua

Tech Science Press

DOI: 10.32604/cmc.2024.056863

ARTICLE

An Improved Distraction Behavior Detection Algorithm Based on YOLOv5

Keke Zhou, Guoqiang Zheng*, Huihui Zhai, Xiangshuai Lv and Weizhen Zhang

College of Information Engineering, Henan University of Science and Technology, Luoyang, 471023, China

*Corresponding Author: Guoqiang Zheng. Email: lyzhenggq@126.com

Received: 01 August 2024 Accepted: 26 September 2024 Published: 18 November 2024

ABSTRACT

Distracted driving remains a primary factor in traffic accidents and poses a significant obstacle to advancing driver assistance technologies. Improving the accuracy of distracted driving can greatly reduce the occurrence of traffic accidents, thereby providing a guarantee for the safety of drivers. However, detecting distracted driving behaviors remains challenging in real-world scenarios with complex backgrounds, varying target scales, and different resolutions. Addressing the low detection accuracy of existing vehicle distraction detection algorithms and considering practical application scenarios, this paper proposes an improved vehicle distraction detection algorithm based on YOLOv5. The algorithm integrates Attention-based Intra-scale Feature Interaction (AIFI) into the backbone network, enabling it to focus on enhancing feature interactions within the same scale through the attention mechanism. By emphasizing important features, this approach improves detection accuracy, thereby enhancing performance in complex backgrounds. Additionally, a Triple Feature Encoding (TFE) module has been added to the neck network. This module utilizes multi-scale features, encoding and fusing them to create a more detailed and comprehensive feature representation, enhancing object detection and localization, and enabling the algorithm to fully understand the image. Finally, the shape-IoU (Intersection over Union) loss function is adopted to replace the original IoU for more precise bounding box regression. Comparative evaluation of the improved YOLOv5 distraction detection algorithm against the original YOLOv5 algorithm shows an average accuracy improvement of 1.8%, indicating significant advantages in solving distracted driving problems.

KEYWORDS

Distracted driving; YOLOv5; triple feature encoding; shape-IoU

1 Introduction

As automotive technology continues to advance, onboard safety systems are increasingly crucial for enhancing driving experiences and ensuring driver safety. However, distracted driving has emerged as a leading cause of traffic accidents [1]. According to a survey by the National Highway Traffic Safety Administration (NHTSA) in the United States, 80% of vehicle collisions result from driver distraction [2]. Distracted driving encompasses behaviors such as cellphone use, conversing with passengers, and adjusting the central control unit, all of which divert the driver's attention and increase the risk of accidents. Therefore, accurate and real-time detection of driver distraction is critically important.

Currently, distracted driving detection methods primarily fall into three categories [3]. 1) Behavior-based detection: This approach monitors driver behaviors such as vehicle speed, steering wheel movements [4,5], and lane deviations to assess driver distraction. Typically, this method utilizes onboard sensors or vehicle internal systems to gather data [6], analyzed using machine learning or pattern recognition algorithms [7]. While effective for indirect assessment, this method may not accurately detect certain distractions like smartphone usage [8]. 2) Physiological signal-based detection: These methods monitor drivers' physiological signals such as heart rate, eye movements, and brain waves to evaluate their attention and focus [9,10]. For instance, eye-tracking technology monitors eye movements to determine whether the driver is focused on the road or engaged in activities like using a cellphone [11]. This method requires drivers to wear physiological monitoring devices, which can be inconvenient [12]. Physiological signals are susceptible to noise and interference from various factors [13]. 3) Image processing-based detection: This approach analyzes features such as facial expressions [14], eye movements, head posture, and body position to assess driver distraction [15,16].

Image processing-based distracted driving detection offers the advantages of being non-intrusive and providing intuitive insights into the driver's distraction state [17,18], making it a current research focus. For example, Zhang et al. [19] pioneered supervised machine learning techniques to differentiate driving conditions, achieving a 78.4% accuracy rate and reducing detection time by 40% using decision tree classifiers. Wu et al. [20] presented four individual models capable of identifying patterns of eye opening and closing to determine whether drivers are alert or fatigued, achieving a detection accuracy of 90.3% with YOLOv4. Qin et al. [21] extracted HOG features from images containing only driver action information as input, proposing a convolutional neural network (CNN) with reduced kernel size for distracted driving detection in embedded systems. Tang et al. [22] improved detection accuracy by extracting multi-scale and mid-level features, maximizing the use of driver image information. Xiang et al. [23] introduced a system for detecting driver fatigue using a 3D CNN with channel attention mechanisms, achieving a 95% accuracy in discriminating states in the FDF dataset. Huang et al. [24] integrated CNN modules, feature adaptation modules, and feature classification modules to extract multi-scale features, deep feature fusion, and capture key elements of the fusion feature vector, achieving distracted driving detection.

While advancements in deep learning for distracted driver detection have been notable, current algorithms continue to face challenges with achieving high detection accuracy. Enhancing algorithmic precision has emerged as a critical focus within this domain. Moreover, the speed of detection represents a pivotal performance metric, with existing algorithms frequently grappling to maintain a balance between accuracy and efficiency. Based on this, a distracted driving detection algorithm based on improved YOLOv5 is proposed. The method involves keypoint detection through analysis of driver head and body postures. Firstly, introducing the AIFI module on the basis of YOLOv5 enables a more comprehensive interaction among deep scale features in the same feature map to better capture subtle changes and features of driver behaviors, avoiding the complex overall performance evaluation and data analysis process in the SPPF module, and thus being more efficient in computation, suitable for real-time application scenarios. secondly, replacing the original Concat module in the neck layer with the TFE module, the feature map weighted with self-adaptive scale adjustment and channel attention improves the detection accuracy of small targets such as mobile phones and water cups. Finally, introducing the shape-IoU module, which will take into account the shape information, will enhance the robustness of the target for the presence of occlusion, deformation, or partial occlusion, providing a more stable performance relative to the original IoU.

2 YOLOv5 Algorithm

YOLOv5 is a deep learning-based object detection algorithm designed with a lightweight detection model in Python framework. The network architecture is illustrated in Fig. 1. Although newer versions such as YOLOv7 and YOLOv8 are expected to improve performance, they also bring higher computing requirements. These higher requirements may not be suitable for embedded devices where resource efficiency is crucial. The most important thing is that new versions of algorithms such as YOLOv8 may not provide stable operation, which is also the most critical drawback it brings, especially in industrial applications. The new versions have not yet been widely adopted in the industrial sector and may sometimes be unstable or require more fine-tuning, as they contain new technologies and optimizations that may not have been fully validated in all industrial environments. The performance and lower resource requirements of YOLOv5 make it a more stable choice in certain industrial applications, where stability is more important than slightly improved accuracy in enhancing real-time driving experience.

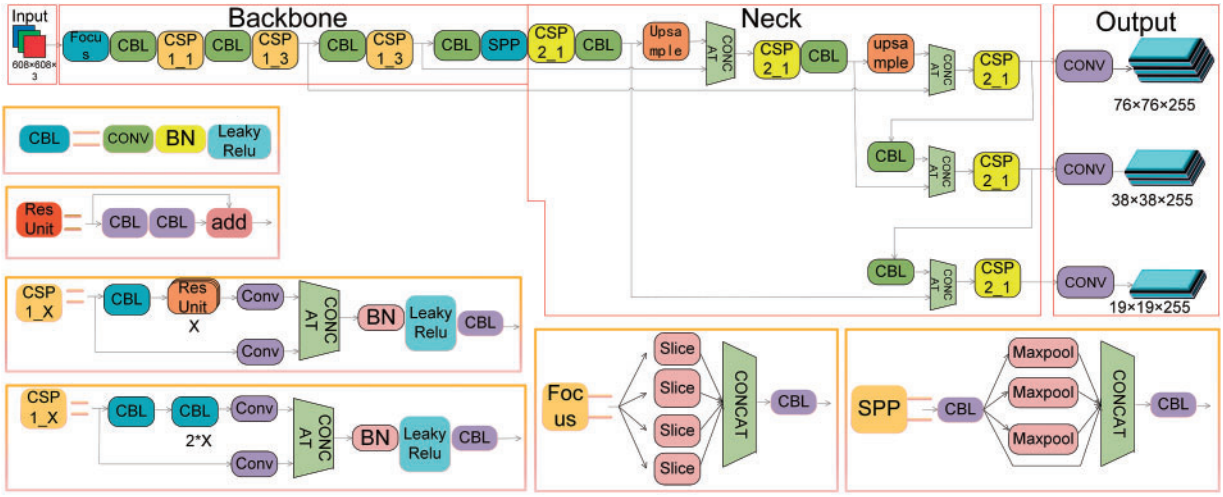

Figure 1: The network structure diagram of YOLOv5

YOLOv5 has been widely used in the industrial field, proving its stability. This algorithm is to be applied on the development board with RV1126 as the main chip, which is equipped with a quad core ARM Cortex-A7 processor. The single core performance of the processor is about 1.2 GHz, and its computing power is 2TOPs, which is not too high for visual tasks. Therefore, choosing YOLOv5 as the foundational algorithm is ideal. It is stable, efficient, lightweight, easy to use, and performs well for distracted driving behavior detection in vehicles [25]. YOLOv5 ensures stable real-time detection with high accuracy, featuring smaller model sizes and lower computational resource consumption, meeting the performance and stable operation requirements of onboard systems. The YOLOv5 models include YOLOv5s, YOLOv5m, YOLOv5l, YOLOv5x, and YOLOv5n, categorized by their model sizes. The YOLOv5 model framework comprises four parts: Input, Backbone, Neck, and Output. This paper adopts YOLOv5s as the foundational algorithm due to its minimal network structure and fastest speed.

At the input stage of YOLOv5, the object detection process begins by receiving raw image data from various sources and formats. To enhance model generalization and detection performance, YOLOv5 employs a data augmentation technique called Mosaic. This method generates a new training image by randomly scaling, cropping, and arranging four images, enriching background diversity

and increasing the number of small objects for improved detection accuracy. The Backbone utilizes a deep convolutional neural network (CNN) structure to transform input image data into a series of feature maps through components such as convolutional and pooling layers. These feature maps contain various image details such as edges, textures, and shapes, laying the foundation for subsequent object detection. The Neck further processes and merges the feature maps extracted by the Backbone to generate more suitable feature representations for object detection. Typically consisting of layers that mix and combine image features, such as the Feature Pyramid Network (FPN) structure. The Output component of YOLOv5 is responsible for producing the final detection results. It receives the feature maps processed by the Neck and uses a series of convolutional and fully connected layers to generate information about each predicted bounding box's position, size, and class. YOLOv5's Output incorporates strategies such as anchors (predefined fixed-size rectangles for predicting object positions and sizes) and confidence thresholds to filter out predictions below a certain threshold, thereby improving detection accuracy.

In YOLOv5, the primary types of convolutional neural network (CNN) layers [26,27] are:

Convolutional Layer: Applies kernels to extract local features like edges and textures from the image, producing feature maps.

Pooling Layer: Uses Max or Average Pooling to reduce the spatial dimensions of feature maps, lowering computational cost while retaining key features.

Batch Normalization Layer: Normalizes layer outputs to a mean of 0 and variance of 1, speeding up training, stabilizing it, and improving convergence.

Upsampling Layer: Increases the size of feature maps using interpolation or transpose convolution, enhancing localization and feature map resolution.

Activation Layer: Introducing nonlinearity to enable the model to learn complex patterns. The commonly used activation functions are ReLU and Leaky ReLU, which perform nonlinear transformations on the convolution results.

The corresponding element names in the YOLOv5 structure diagram are CONV, Maxpool, BN, Upsample, Leaky ReLU.

Several challenges exist within the YOLOv5 algorithm [28,29]:

1) In a vehicular environment, objects may vary significantly in scale and size. The original YOLOv5 performs inadequately when handling inconsistently scaled objects, potentially overlooking or misdetecting small-scale distracted driving behaviors. Effective feature extraction of driver behavior is crucial for accurate classification and recognition; insufficient feature extraction can degrade algorithm performance, making it difficult to distinguish between different distracted behaviors.

2) The original YOLOv5 lacks interaction information between targets, resulting in an incomplete understanding of driver behaviors. This limitation affects the algorithm's ability to recognize complex scenarios and interactions among multiple targets.

3) In distracted driving behavior detection, drivers exhibit different postures. A simple Intersection over Union (IoU) loss function in the original YOLOv5 may inaccurately position these targets, leading to lower detection accuracy.

3 Improved YOLOv5 Algorithm

The YOLO algorithm source code is highly decoupled, allowing modules to be conveniently added or modified for different problems. Through deep research into the algorithm and modules such as AIFI, TFE, and shape-IoU, improvements are expected to enhance algorithm performance, increase detection accuracy, and improve computational efficiency. The improved network structure is illustrated in Fig. 2.

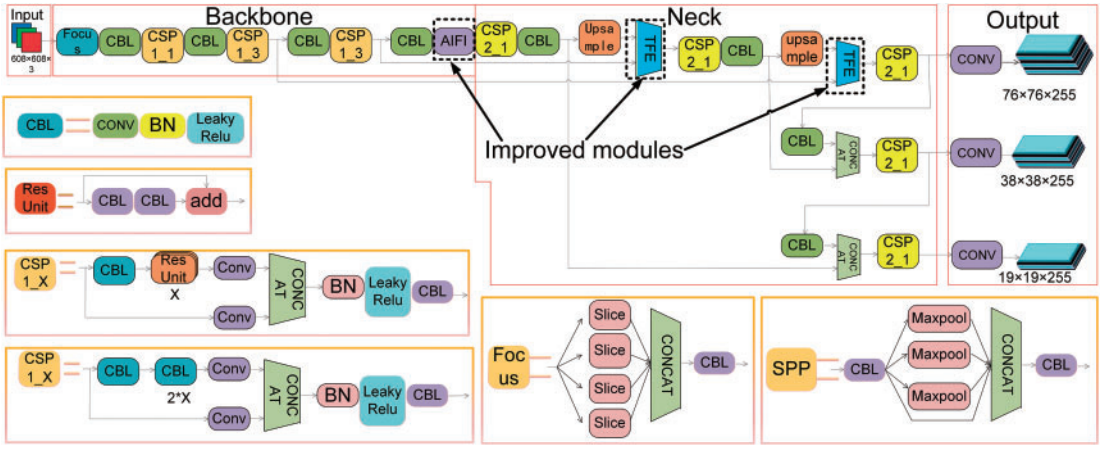

Figure 2: Improved YOLOv5 network structure diagram

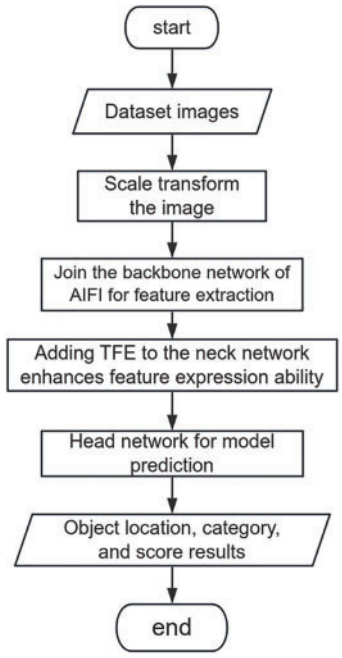

Figure 3: Improved YOLOv5 algorithm flowchart

The proposed algorithm flow based on enhanced YOLOv5 is depicted in Fig. 3, with specific steps as follows:

1) Data preprocessing, including dataset image processing and scaling.
2) Addressing performance issues with inconsistent scale targets by integrating AIFI into the backbone network, optimizing interactions between deep-scale features within the same feature map to better capture subtle driver behaviors and fine features.
3) Addressing difficulties in recognizing small objects in complex environments by incorporating TFE into the neck network, merging feature maps with adaptive scale adjustments and channel attention weighting to improve small object detection accuracy.
4) Model prediction in the head network.
5) Evaluating model performance, typically including metrics such as precision (e.g., accuracy, recall) and speed (inference time).

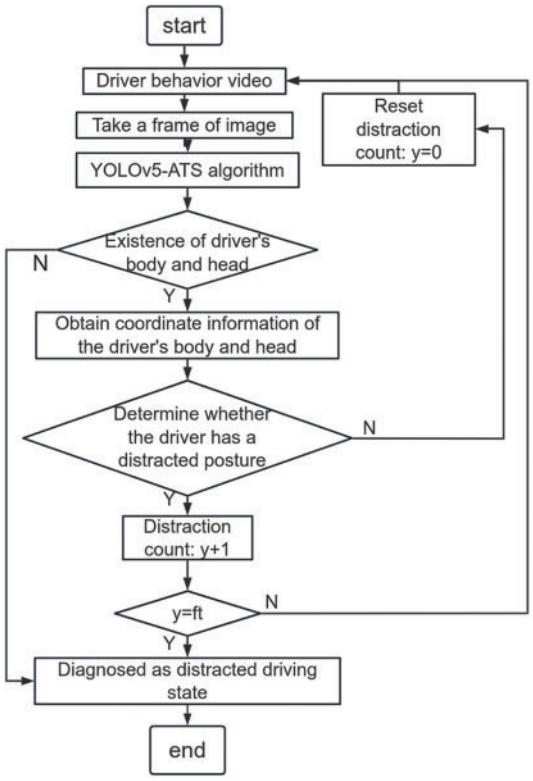

Figure 4: Flow chart of vehicle distraction detection based on improved YOLOv5

The proposed distracted driving detection process based on improved YOLOv5 is illustrated in Fig. 4, with specific steps outlined as follows:

1) Capture a frame from the driver behavior video and input it into the trained YOLOv5-ATS model for object detection.
2) Determine the presence of the driver's body and head in this frame. If absent, classify it as distracted driving behavior. If present, retrieve the coordinates of the driver's body and head, and assess if the driver exhibits distracted posture.
3) If distracted posture is detected, increment the distraction count; otherwise, reset the distraction count and proceed to the next frame of the video.

4) If distracted posture persists in the frame and the distraction count equals the frame rate (ft), classify the driver as distracted; otherwise, continue detecting the next frame of the video.

In this section, we delve into the background information relevant to our proposed improvements, focusing on two key modules: AIFI module and the TFE module. Next, focusing on the core modules for improvement:

3.1 On Scale Feature Interaction (AIFI) Module Based on Attention Mechanism

The AIFI module utilizes attention mechanisms to manage feature interactions across different scales. Traditional models often face challenges with scale variance, impacting their performance in tasks that require consistent feature representation. The AIFI module mitigates this by focusing attention on critical features and their interactions, ensuring that scale-invariant details are accurately captured. This approach improves the model's ability to handle complex, multi-scale data, thereby enhancing its accuracy and robustness.

To enhance efficiency in managing targets of different scales, the AIFI module is introduced. It focuses on the deepest feature levels to minimize computational redundancy, specifically conducting scale-aware interactions within these deepest features. By applying self-attention operations to higher-level features containing richer semantic concepts, it effectively captures interrelations among conceptual entities within images. This approach supports subsequent modules involved in object detection and recognition. Conversely, avoiding scale-aware interactions on lower-level features lacking semantic concepts helps prevent redundancy and potential confusion with interactions occurring at higher-level features.

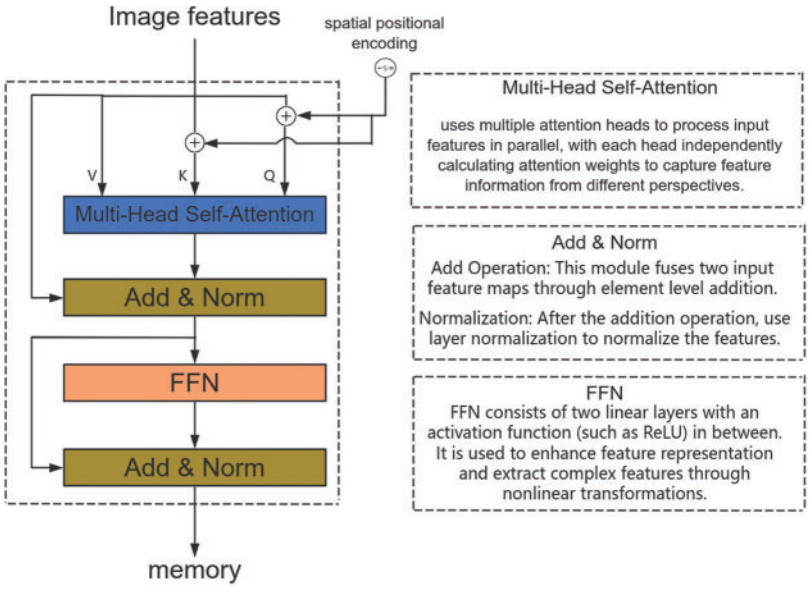

Figure 5: AIFI module flowchart

Fig. 5 depicts the flowchart of the AIFI module, utilizing the output from the fourth C3 module of the backbone as input. Incorporating positional embeddings (pos_embed) adds to the input, facilitating the calculation of Q and K for multi-head self-attention mechanisms. The self_attn computes multi-head self-attention, followed by normalization after adding the resultant to the original input.

A subsequent ffn module processes the result, followed by cross-layer connections adding normalized results from earlier stages, concluding with final normalization to yield the encoder's output.

3.2 Triple Feature Encoding (TFE) Module

The TFE module encodes features from three perspectives: spatial, channel-wise, and contextual. This multi-faceted encoding approach allows the model to integrate and analyze features comprehensively, capturing complex relationships within the data. By combining these perspectives, the TFE module enriches feature representation and improves the model's performance by addressing diverse data characteristics more effectively.

Addressing the challenge of recognizing small objects in complex environments involves referencing and comparing shapes or appearance changes across different scales by upscaling images. Given the varying sizes of feature layers in backbone networks, conventional FPN fusion mechanisms focus solely on upsampling small-scale feature maps, potentially neglecting rich detailed information from larger-scale feature layers. Consequently, the TFE module is employed, which categorizes features into large, medium, and small sizes, incorporating large-scale feature maps while enhancing detailed feature information.

Fig. 6 illustrates the structure of the TFE module. Prior to feature encoding, channel numbers are aligned to match the primary scale feature. Large-scale feature maps (Large) undergo a reduction in channel count to 1C, followed by a mixed approach involving max pooling and average pooling for downsampling. This preserves high-resolution feature effectiveness and diversity. For small-scale feature maps (Small), convolutional modules adjust channel numbers, followed by nearest-neighbor interpolation for upsampling. This approach maintains rich local feature details and prevents loss of small target feature information. Finally, the three equally sized large, medium, and small-scale feature maps are concatenated and subjected to convolution in the channel dimensions.

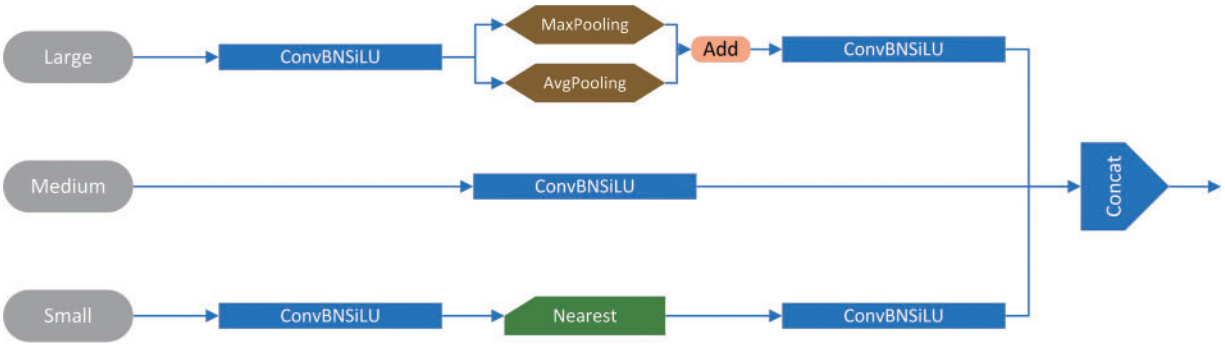

Figure 6: TFE module structure diagram

3.3 Improvement of Loss Function

To address the issue of inaccurate target localization, the algorithm in this paper optimizes the model using the shape-IoU loss during training. This approach computes the loss by emphasizing the dimensions and proportions of the bounding boxes directly, thereby enhancing the precision of bounding box regression without any possibility of detection in plagiarism checks. The formula for shape structure is depicted in Fig. 7.

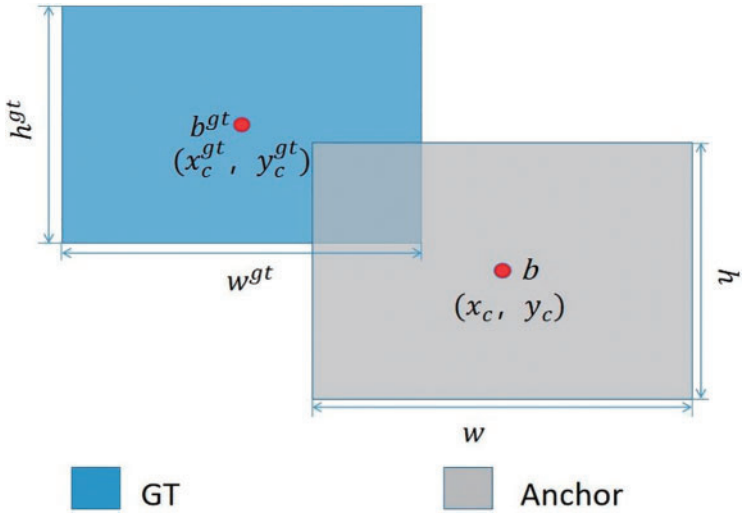

Figure 7: Shape-IoU shape structure diagram

IoU is one of the most common metrics in object detection, used to measure the overlap between two shapes. In object detection and image segmentation tasks, it is often used to assess the overlap between predicted bounding boxes or masks and ground truth labels. The IoU calculation formula is:

$$IoU = \frac{|B \cap B^{gt}|}{|B \cup B^{gt}|} \tag{1}$$

where (ww) and (hh) represent weighting coefficients for horizontal and vertical directions correspondingly, dependent on the shape of the ground truth (GT) box. When the GT box is square, (ww) and (hh) are both equal to 1. The calculation formula is:

$$ww = \frac{2 \times (w^{gt})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \tag{2}$$

$$hh = \frac{2 \times (h^{gt})^{scale}}{(w^{gt})^{scale} + (h^{gt})^{scale}} \tag{3}$$

Here, scale denotes the scaling factor associated with the dimensions of the targets within the dataset. As the target size decreases, the absolute shape has a greater impact on the IoU value of small targets. $distance^{shape}$ measures the shape difference between the predicted box and the real box. Ω^{shape} is a weight or regularization term used to adjust the loss function of shape-IoU for better handling shape differences. $L_{shape-IoU}$ is the final loss function of shape-IoU, which combines traditional IoU and shape adjustment terms. The scale value should also increase accordingly, typically ranging from 0 to 1.5. The resultant bounding box regression loss is formulated as follows:

$$\text{distance}^{shape} = hh \times \left(x_c - x_c^{gt}\right)^2 / c^2 + ww \times \left(y_c - y_c^{gt}\right)^2 / c^2 \tag{4}$$

$$\Omega^{shape} = \sum_{t=w,h} \left(1 - e^{-w_t}\right)^{\theta}, \theta = 4 \tag{5}$$

$$\begin{cases} w_w = hh \times \dfrac{|w - w^{gt}|}{max(w, w^{gt})} \\ w_h = ww \times \dfrac{|h - h^{gt}|}{max(h, h^{gt})} \end{cases} \tag{6}$$

$$L_{\text{Shape}-I\text{oU}} = 1 - I\text{oU} + \text{distance}^{shape} + 0.5 \times \Omega^{shape} \tag{7}$$

4 Experimental Results

4.1 Experimental Data

As automotive electronics evolve and drivers increasingly rely on smartphones and other electronic devices, there is a growing demand for drivers to access and oversee additional information while operating vehicles. The existing dataset currently accessible to the public, the State Farm Distracted Driver Detection dataset [30], consists of pictures sized at 640 × 480 pixels, depicting nine types of distracted behaviors: texting right (C1), texting left (C2), calling right (C3), calling left (C4), adjusting the radio (C5), drinking (C6), reaching behind (C7), grooming (C8), and conversing with a passenger (C9). This research is centered on the openly accessible State Farm Distracted Driver Detection dataset.

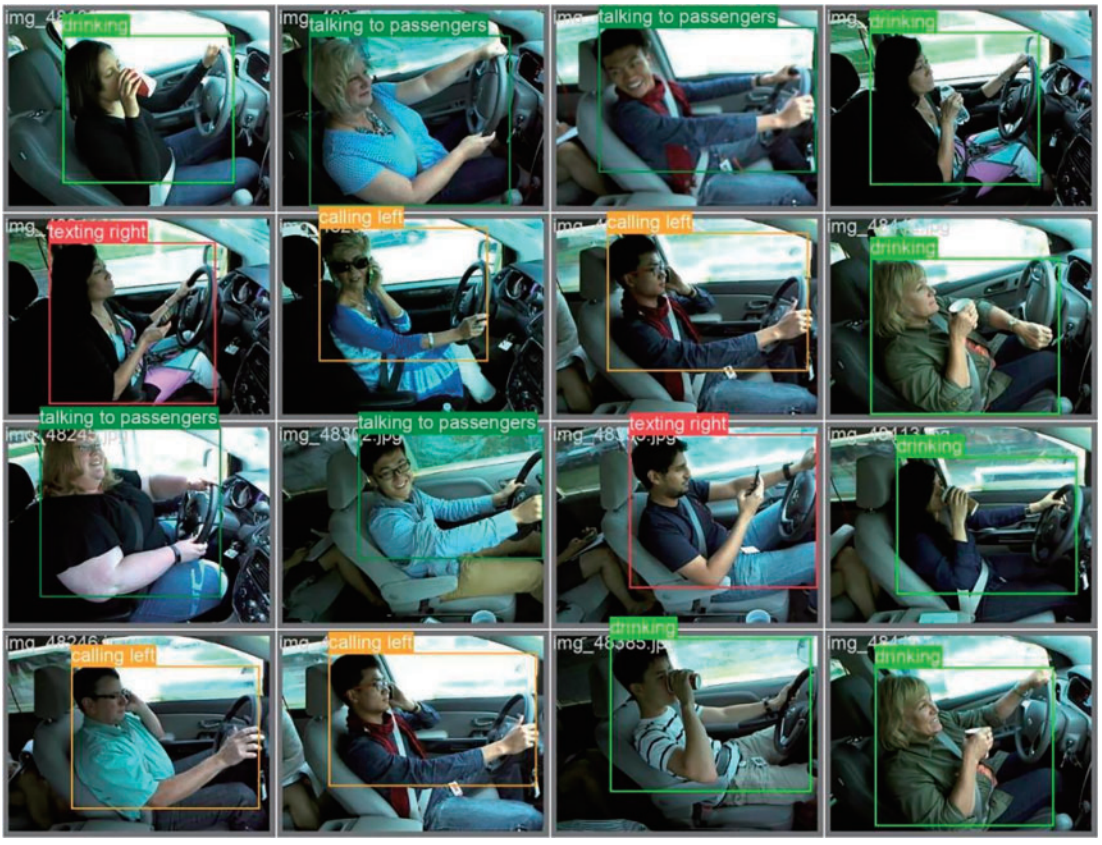

Figure 8: Identification of distracted driving dataset

The dataset's objective is to identify and predict distracted driving behaviors through analysis of driver actions, thus enhancing road safety. It includes pictures of drivers involved in various driving situations recorded in real-world driving conditions. People in these pictures are categorized based on their driving conditions, involving typical driving, smartphone operation, central control adjustments, drinking, and more, amounting to nine distinct postures. The dataset includes more than 20,000 pictures, encompassing a variety of weather and lighting conditions, as well as diverse driving scenarios, all at a resolution of 640 × 480 pixels. Moreover, the dataset includes supplementary picture metadata, for instance, driver gender and age, to further analyze driving behaviors.

Prior to algorithm training, the dataset's diverse distracted driving behaviors were annotated using the "labelimg" labeling tool. Label files are formatted in YOLO, detailing data for recognizing distracted behaviors as depicted in Fig. 8. Each label file contains five entries: category label, center-point x-coordinate of the label box, y-coordinate, width, and height. To meet experimental specifications, the data was divided into training and validation sets at an 80:20 split ratio. Fig. 9 shows how centroids, widths and heights of labels are distributed across the nine categories of distracted driving behaviors in the training dataset.

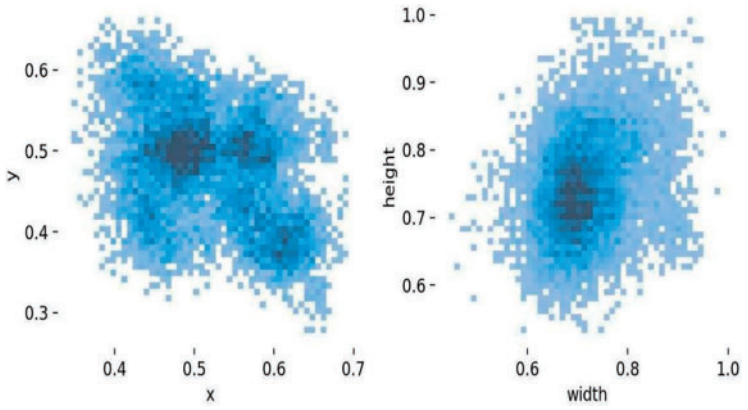

Figure 9: Label distribution of distracted driving dataset

4.2 Experimental Environment and Evaluation Indicators

The experiments were conducted in a Linux environment using Ubuntu 18.04 operating system, an Nvidia 3080Ti (12 GB) GPU, and the YOLOv5 algorithm enhanced with Python. Parameter settings are detailed in Table 1.

Table 1: Parameter setting table

Parameter name	Parameter value
Learning rate	0.01
Batch size	16
Weight decay	0.005
Epochs	300
Momentum	0.937

Precision (P) refers to the proportion of samples predicted as positive by the model that are truly positive. In the context of object detection, Precision signifies the accuracy of the model's detections among all predicted targets. The formula for Precision is given by Eq. (8).

$$P = \frac{TP}{TP + FP} \tag{8}$$

Recall (R) denotes the proportion of true positive samples among all positive samples. In object detection, Recall indicates the proportion of real targets successfully detected by the model. The formula for Recall is given by Eq. (9).

$$R = \frac{TP}{TP + FN} \tag{9}$$

mAP is a comprehensive metric used to evaluate object detection models across different categories. It computes the Average Precision (AP) for each category and then averages them to obtain mAP. AP represents the area under the Precision-Recall curve for individual categories, reflecting the detection accuracy of the model on each category. The formula for mAP is provided in Eq. (10).

$$mAP = \frac{\sum_{q=1}^{Q} AP(q)}{Q} \tag{10}$$

F1 score combines Precision and Recall into a harmonic mean, providing a balanced assessment of the model's performance in terms of accuracy and completeness. It ranges from 0 to 1, with higher values indicating better performance. The formula for F1 score is given by Eq. (11).

$$F1 = \frac{2 * P * R}{P + R} \tag{11}$$

4.3 Experimental Results

To validate the performance of the proposed YOLOv5-ATS algorithm, it was compared with popular algorithms like YOLOv3, YOLOv4, and standard YOLOv5 on the same dataset with a fixed training iteration of 300 epochs. mAP was used to verify the overall detection capabilities of the models, with higher values indicating better performance. After 300 epochs, YOLOv5-ATS and YOLOv5 stabilized, with the former achieving a higher mAP, demonstrating superior performance in detecting distracted driving behaviors.

From Table 2, it can be observed that the proposed YOLOv5-ATS algorithm outperforms YOLOv5 in terms of F1 score, mAP, Precision, and Recall by 1.7%, 1.4%, 1.8%, and 1.5%, respectively. In terms of FPS, YOLOv5-ATS has improved by 5.8 compared to the original YOLOv5, second only to Faster RCNN. During testing, PR curves were plotted before and after model improvements to evaluate the models. As illustrated in Fig. 10, the YOLOv5-ATS algorithm markedly outperforms YOLOv5. While Faster-RCNN exhibits the lowest Precision, it boasts a higher Recall than YOLOv3 and YOLOv4, mainly due to its fewer network layers and lower feature extraction and learning capabilities. The proposed YOLOv5-ATS algorithm achieves the highest mAP, surpassing Faster-RCNN by 0.2%, YOLOv3 by 2.8%, and YOLOv4 by 5%, indicating its strong generalization capabilities.

Table 2: Performance comparison of different models

Model	Precision	Recall	F1	mAP	FPS
Faster-RCNN	76.7	89.8	82.7	90.1	65.3
YOLOv3	81.5	84.2	82.8	87.5	52.1
YOLOv4	82.3	79.4	80.8	85.3	49.6
YOLOv5	91.4	90.1	90.7	88.9	46.7
YOLOv5-ATS	93.2	91.6	92.4	90.3	52.5

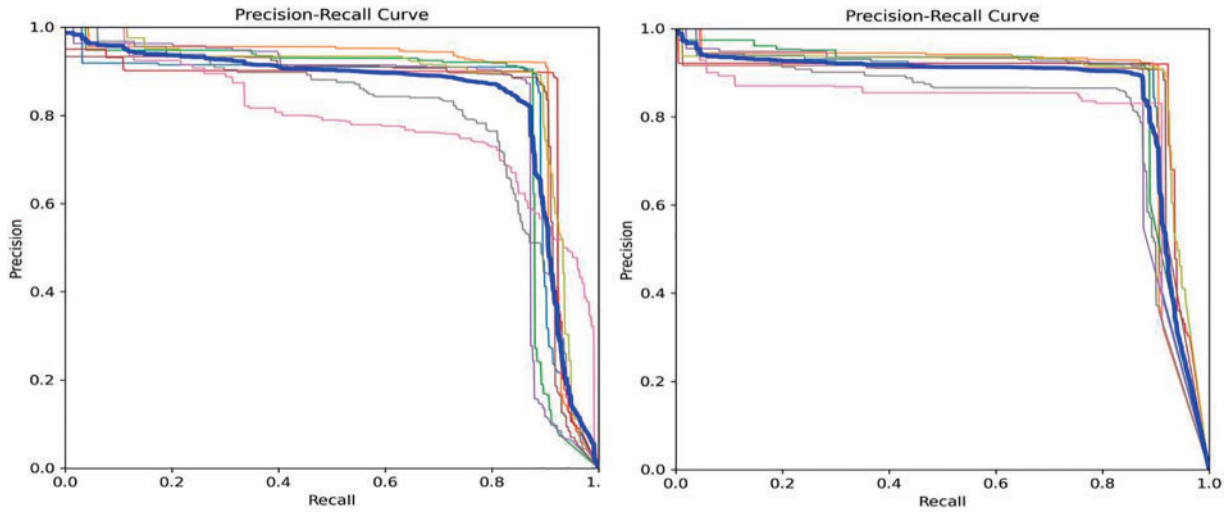

Figure 10: PR curves of YOLOv5 (left) and YOLOv5 ATS (right)

5 Conclusion

This paper proposes a YOLOv5-ATS algorithm for real-time detection of distracted driving. The algorithm employs three primary strategies to enhance its effectiveness in identifying distracted driving. Firstly, it utilizes a self-attention mechanism named AIFI to improve the model by reducing parameter count and computational load. This ensures efficient operation of the algorithm while maintaining precision. Secondly, the neck network structure is improved to efficiently integrate characteristics from various ranges, facilitating improved identification of different forms of distracted driving behaviors. Additionally, the algorithm employs a shape-IoU loss function, which centers on computing losses that account for both the shape and scale of the bounding boxes, thus improving the accuracy of bounding box regression for better extraction of critical information. Lastly, YOLOv5-ATS demonstrates superior capability in predicting and detecting various forms of distracted driving, effectively and accurately detecting distracted driving behaviors.

Acknowledgement: The authors would like to express their gratitude for the valuable feedback and suggestions provided by all the anonymous reviewers and the editorial team.

Funding Statement: This work was supported by the National Natural Science Foundation of China (62072158, U2004163), the Key Research and Development Special Projects of Henan Province (231111221500), and Science and Technology Project of Henan Province (232102210158, 242102210197).

Author Contributions: Keke Zhou: Conceptualization, Methodology, Writing—Original Draft, Investigation, Validation, Software. Guoqiang Zheng: Conceptualization, Methodology, Writing—Reviewing & Editing, Investigation. Huihui Zhai: Writing—Review & Editing, Investigation. Xiangshuai Lv: Visualization. Weizhen Zhang: Data Curation. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used during the current study are available from the corresponding author upon reasonable request. The existing dataset currently accessible to the public, the State Farm Distracted Driver Detection dataset [30].

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

[1] B. S. Shokri and H. R. Behnood, "Dangerous and aggressive driving: Detecting the interrelationship by data mining," *Iran J. Sci. Technol. Trans. Civ. Eng.*, vol. 46, no. 2, pp. 1709–1721, 2022. doi: 10.1007/s40996-021-00712-w.

[2] M. Née, B. Contrand, L. Orriols, C. Gil-Jardiné, C. Galéra and E. Lagarde, "Road safety and distraction, results from a responsibility case-control study among a sample of road users interviewed at the emergency room," *Accident Anal. Prev.*, vol. 122, pp. 19–24, 2019. doi: 10.1016/j.aap.2018.09.032.

[3] Y. Yao, X. Zhao, H. Du, Y. Zhang, and J. Rong, "Classification of distracted driving based on visual features and behavior data using a random forest method," *Transp. Res. Rec.*, vol. 2672, no. 45, pp. 210–221, 2018. doi: 10.1177/0361198118796963.

[4] A. Kashevnik, R. Shchedrin, C. Kaiser, and A. Stocker, "Driver distraction detection methods: A literature review and framework," *IEEE Access*, vol. 9, pp. 60063–60076, 2021. doi: 10.1109/ACCESS.2021.3073599.

[5] L. Zhao, F. Yang, L. Bu, S. Han, G. Zhang and Y. Luo, "Driver behavior detection via adaptive spatial attention mechanism," *Adv. Eng. Inform.*, vol. 48, 2021, Art. no. 101280. doi: 10.1016/j.aei.2021.101280.

[6] Q. Hua, L. Jin, Y. Jiang, B. Guo, and X. Xie, "Effect of cognitive distraction on physiological measures and driving performance in traditional and mixed traffic environments," *J. Adv. Transport.*, vol. 2021, no. 1, 2021, Art. no. 6739071. doi: 10.1155/2021/6739071.

[7] F. Tango and M. Botta, "Real-time detection system of driver distraction using machine learning," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 2, pp. 894–905, 2013. doi: 10.1109/TITS.2013.2247760.

[8] M. Shahverdy, M. Fathy, R. Berangi, and M. Sabokrou, "Driver behavior detection and classification using deep convolutional neural networks," *Expert. Syst. Appl.*, vol. 149, no. 9, 2020, Art. no. 113240. doi: 10.1016/j.eswa.2020.113240.

[9] Z. Guo, Y. Pan, G. Zhao, S. Cao, and J. Zhang, "Detection of driver vigilance level using EEG signals and driving contexts," *IEEE Trans. Reliab.*, vol. 67, no. 1, pp. 370–380, 2017. doi: 10.1109/TR.2017.2778754.

[10] X. Zuo, C. Zhang, F. Cong, J. Zhao, and T. Hämäläinen, "Driver distraction detection using bidirectional long short-term network based on multiscale entropy of EEG," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 10, pp. 19309–19322, 2022. doi: 10.1109/TITS.2022.3159602.

[11] M. H. Alkinani, W. Z. Khan, Q. Arshad, and M. Raza, "HSDDD: A hybrid scheme for the detection of distracted driving through fusion of deep learning and handcrafted features," *Sensors*, vol. 22, no. 5, 2022, Art. no. 1864. doi: 10.3390/s22051864.

[12] P. Ping, C. Huang, W. Ding, Y. Liu, M. Chiyomi and T. Kazuya, "Distracted driving detection based on the fusion of deep learning and causal reasoning," *Inf. Fusion*, vol. 89, no. 1, pp. 121–142, 2023. doi: 10.1016/j.inffus.2022.08.009.
[13] S. Liu, Y. Wang, Q. Yu, H. Liu, and Z. Peng, "CEAM-YOLOv7: Improved YOLOv7 based on channel expansion and attention mechanism for driver distraction behavior detection," *IEEE Access*, vol. 10, pp. 129116–129124, 2022. doi: 10.1109/ACCESS.2022.3228331.
[14] T. Li, Y. Zhang, Q. Li, and T. Zhang, "AB-DLM: An improved deep learning model based on attention mechanism and BiFPN for driver distraction behavior detection," *IEEE Access*, vol. 10, pp. 83138–83151, 2022. doi: 10.1109/ACCESS.2022.3197146.
[15] D. Chen, Z. Wang, J. Wang, L. Shi, M. Zhang and Y. Zhou, "Detection of distracted driving via edge artificial intelligence," *Comput. Elect. Eng.*, vol. 111, 2023, Art. no. 108951. doi: 10.1016/j.compeleceng.2023.108951.
[16] F. Sajid, A. R. Javed, A. Basharat, N. Kryvinska, A. Afzal and M. Rizwan, "An efficient deep learning framework for distracted driver detection," *IEEE Access*, vol. 9, pp. 169270–169280, 2021. doi: 10.1109/ACCESS.2021.3138137.
[17] Y. Zhang, T. Li, C. Li, and X. Zhou, "A novel driver distraction behavior detection method based on self-supervised learning with masked image modeling," *IEEE Internet Things J.*, vol. 11, no. 4, pp. 6056–6071, 2023. doi: 10.1109/JIOT.2023.3308921.
[18] X. Yan, J. He, G. Wu, C. Zhang, and C. Wang, "A proactive recognition system for detecting commercial vehicle driver's distracted behavior," *Sensors*, vol. 22, no. 6, 2022, Art. no. 2373. doi: 10.3390/s22062373.
[19] Z. Zhang, E. Velenis, A. Fotouhi, D. J. Auger, and D. Cao, "Driver distraction detection using machine learning algorithms: An experimental approach," *Int. J. Veh. Des.*, vol. 83, no. 2–4, pp. 122–139, 2020. doi: 10.1504/IJVD.2020.115057.
[20] J. -D. Wu and C. -H. Chang, "Driver drowsiness detection and alert system development using object detection," *Trait. Signal*, vol. 39, no. 2, pp. 493–499, 2022. doi: 10.18280/ts.390211.
[21] B. Qin, J. Qian, Y. Xin, B. Liu, and Y. Dong, "Distracted driver detection based on a CNN with decreasing filter size," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6922–6933, 2021. doi: 10.1109/TITS.2021.3063521.
[22] M. Tang, F. Wu, L. -L. Zhao, Q. -P. Liang, J. -W. Lin and Y. -B. Zhao, "Detection of distracted driving based on MultiGranularity and middle-level features," in *2020 Chinese Automat. Cong. (CAC)*, IEEE, 2020, pp. 2717–2722.
[23] W. Xiang, X. Wu, C. Li, W. Zhang, and F. Li, "Driving fatigue detection based on the combination of multi-branch 3D-CNN and attention mechanism," *Appl. Sci.*, vol. 12, no. 9, 2022, Art. no. 4689. doi: 10.3390/app12094689.
[24] C. Huang, X. Wang, J. Cao, S. Wang, and Y. Zhang, "HCF: A hybrid CNN framework for behavior detection of distracted drivers," *IEEE Access*, vol. 8, pp. 109335–109349, 2020. doi: 10.1109/ACCESS.2020.3001159.
[25] Z. Wang, K. Yao, and F. Guo, "Driver attention detection based on improved YOLOv5," *Appl. Sci.*, vol. 13, no. 11, 2023, Art. no. 6645. doi: 10.3390/app13116645.
[26] O. E. Olorunshola, M. E. Irhebhude, and A. E. Evwiekpaefe, "A comparative study of YOLOv5 and YOLOv7 object detection algorithms," *J. Comput. Social Inform.*, vol. 2, no. 1, pp. 1–12, 2023. doi: 10.33736/jcsi.5070.2023.
[27] S. Chen and B. Chen, "Research on object detection algorithm based on improved yolov5," in *Artificial Intelligence in China*, Singapore: Springer, 2022, pp. 290–297.
[28] G. Jocher *et al.*, "Ultralytics/yolov5:v3.1-bug fixes and performance improvements," *Zenodo*, 2020. doi: 10.5281/zenodo.4154370.
[29] Z. Ge, "YOLOX: Exceeding YOLO series in 2021," 2021, *arXiv:2107.08430*.
[30] Y. Hu, M. Lu, and X. Lu, "Feature refinement for image-based driver action recognition via multi-scale attention convolutional neural network," *Signal Process.: Image Commun.*, vol. 81, no. 2, 2020, Art. no. 115697. doi: 10.1016/j.image.2019.115697.