COLLISION DETECTION USING IMAGE RECOGNISTION AND MACHINE LEARNING

Seminar Report

Submitted in partial fulfilment of the requirements for the award of the degree of

Master Of Science

in

Automotive Embedded System

by

Avin Joseph (*Reg No:* **MGCER202305**)

ESIGELEC
ÉCOLE D'INGÉNIEURS-ES GÉNÉRALISTES

MUTHOOT GLOBAL CENTRE FOR EDUCATION AND RESEARCH, INDIA

MUTHOOT INSTITUTE OF TECHNOLOGY AND SCIENCE

Ernakulam - 682 308

December 2023

Muthoot
Institute of Technology & Science

CERTIFICATE

This is to certify that the report entitled " **COLLISION DETECTION USING IMAGE RECOGNISTION AND MACHINE LEARNING** " is a bonafide record of the **Project** presented by **Avin Joseph**(Reg. No.: **MGCER202305**), in partial fulfilment of the requirements for the award of the degree of **Master Of Science** in **Automotive Embedded System**.

Prof.Prathibha Sudhakaran	**Ms Ancy Joy**	**Dr.Shoba Gopalakrishnan**
	Dr. Sivaprasad	
(Project Guid) *Dept. of ECE*	(Seminar Coordinators) *Dept. of ECE and EEE*	(Academic Coordinator) *Dept. of ECE*

Place :Varikoli
Date : December 2023

ACKNOWLEDGEMENT

I am grateful to almighty who has blessed us with good health, committed and continuous interest throughout the project. .

I would like to thank our Principal Dr Neelakantan P. C for providing me with all the necessary facilities and support. .

I express my sincere gratitude to Asst. Prof. Dr. Shobha Gopalakrishnan for her support and motivation.

I am grateful to our Seminar coordinators Asst. Prof. Ms Ancy Joy and Asst. Prof. Dr. Sivaprasad for critically assessing our work and for giving us valuable suggestions.

I express my sincere thanks to my guide Guide's name for his/her continuous encouragement, invaluable guidance, motivation and enthusiasm throughout my project. I appreciate his sincere help in terms of patience, time and ideas to make my project experience stimulating and productive.

We feel a deep sense of gratitude for the entire teaching and non-teaching staffs.

The last but not the least, We extend our sincere thanks to our parents and friends.

Avin Joseph

ABSTRACT

The seminar explores the innovative realm of collision detection through the integration of image recognition and machine learning technologies. In a world increasingly dependent on automation and autonomous systems, ensuring the safety of these entities is paramount. This seminar delves into the development and application of advanced algorithms that leverage image recognition to detect potential collisions in various scenarios. Machine learning techniques enhance the system's ability to adapt and improve its accuracy over time, making it a dynamic and efficient solution. The discussion will encompass the theoretical foundations, practical implementations, and potential applications of collision detection using image recognition and machine learning, shedding light on the promising advancements that could redefine safety standards across diverse domains, from autonomous vehicles to industrial automation.

With a primary focus on Convolutional Neural Networks (CNNs). The imperative need for robust collision detection systems in diverse applications, ranging from autonomous vehicles to industrial automation, has propelled the integration of advanced technologies. Leveraging the power of CNNs, this seminar delves into the intricate process of teaching machines to recognize and interpret images, thereby enhancing their ability to detect potential collisions accurately. By harnessing the principles of machine learning, our approach aims to develop intelligent systems that can autonomously assess complex visual scenarios in real-time, providing a significant leap forward in safety and efficiency. The seminar navigates through the underlying methodologies, challenges, and potential applications of Collision Detection using Image Recognition and Machine Learning, offering valuable insights into the transformative potential of this interdisciplinary field.

CONTENTS

LIST OF ABBREVATIONS

FR-CNN	Fast Region based Convolutional Neural Network
YOLO	You Only Look Once
CNN	Convolutional Neural Network
CUDA	Compute Unified Device Architecture
GPU	Graphics Processing Unit
VANET	Vehicle Adhoc Network
RFID	Radio Frequency Identification
VATSC	Vehicle Based Traffic Signal Control
TST	Traffic Signal Time
LED	Light Emitting Diode
IRIS	Intelligent Roadway Information System
ATMS	Advanced Traffic Management System
CCTV	Closed Circuit Television

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 accidents Condition Worldwide

Over 1.3 million deaths happen each year from road accidents, with a further of about 25 to 65 million people suffering from mild injuries as a result of road accidents. In a survey conducted by the World Health Organisation (WHO) on road accidents based on the income status of the country, it is seen that low and middle-income or developing countries have the highest number of road accident related deaths. Developing countries have road accident death rate of about 23.5 per 100,000 population, which is much higher when compared to the 11.3 per 100,000 population for high-income or developed countries [1]. Over 90deaths happen in developing countries, even though these countries have only half of the world's vehicles. In India, a reported 13 people are killed every hour as victims to road accidents across the country. However, the real case scenario could be much worse as many accident cases are left unreported. With the present data, India is on the way to the number one country in deaths from road accidents due to the poor average record of 13 deaths every hour, which is about 140,000 per year [2]. An accident usually has three phases in which a victim can be found...

Figure 1.1: Accident on a Highway

1.2 General Conditions

In an era dominated by technological advancements, the integration of artificial intelligence and computer vision has revolutionized various industries, presenting innovative solutions to complex challenges. One such paramount challenge is the accurate detection of potential collisions in dynamic environments, a critical aspect for ensuring safety in autonomous systems, industrial automation, and beyond. This seminar delves into the realm of Collision Detection using Image Recognition and Machine Learning, where the amalgamation of these two powerful technologies promises to redefine the landscape of safety and efficiency.

The motivation behind this exploration stems from the increasing prevalence of autonomous vehicles, robotics, and automated systems in our daily lives. As these systems become more integral to our infrastructure, the need for reliable collision detection mechanisms becomes paramount. Traditional methods often fall short in addressing the nuances of dynamic and unpredictable scenarios. Image recognition, coupled with machine learning, provides a promising avenue to overcome these limitations by endowing machines with the ability to interpret visual data in a manner akin to human perception

At the heart of our exploration lies the utilization of Convolutional Neural Networks (CNNs), a specialized class of deep learning algorithms designed for image processing tasks. CNNs have demonstrated exceptional prowess in image recognition and pattern detection, making them ideal candidates for enhancing collision detection capabilities. This seminar aims to unravel the intricacies of training machines to 'see' and interpret visual cues, enabling them to make split-second decisions that can avert potential collisions.

Through this seminar, we embark on a journey to understand the foundational principles, methodologies, and real-world applications of Collision Detection using Image Recognition and Machine Learning. By shedding light on the synergies between these technologies, we aim to contribute to the evolving landscape of intelligent systems, where safety and efficiency converge through the lenses of advanced computer vision and artificial intelligence.

CHAPTER 2

REVIEW OF LITERATURE

2.1 Accident Detection Using Convolutional Neural Networks 2019[1]

In this paper the authors proposed vehicle accident detection system which can track an accident at its moment of occurrence and sends an instantaneous alert SMS regarding the accident to the nearby hospitals and police stations which includes details like timestamp and the geographical location. Unlike other systems in use, which consists of expensive sensors and unwanted hardware, the proposed system is much more cost effective and foolproof with a much-improved accuracy rate than its counterparts mainly due to a model-based approach. The experimentation, testing and validation has been carried out using images and the results show that higher sensitivity and accuracy is indeed achieved using this method, henceforth, making it a viable option for implementing this system in most of the state and national highways of the country. Thus, the project works towards a social cause and helps create a system which guarantees that no individual is left unattended or helpless in an unforeseen event of an accident, in turn, securing and maintaining the quality of life to the highest standards.

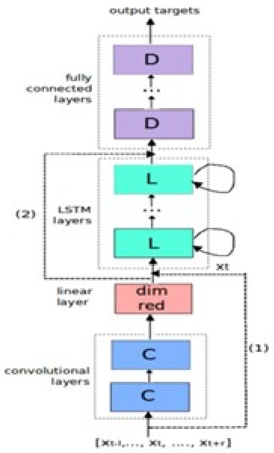


Figure 2.1: Architecture of proposed model

2.2 An Automatic Car Accident Detection Method Based on Cooperative Vehicle Infrastructure Systems[2]

In this paper, the authors have proposed an automatic car accident detection method based on CVIS. First of all, we present the application principles of our proposed method in the CVIS. Secondly, we build a novel image dataset CAD-CVIS, which is more suitable for car accident detection method based on intelligent roadside devices in CVIS. Then we develop the car accident detection model YOLO-CA based on CAD-CVIS and deep learning algorithms. In the model, we combine the multi-scale feature fusion and loss function with dynamic weights to improve real-time and accuracy of YOLO-CA. Finally, we show the simulation experiments results of our method, which demonstrates our proposed methods can detect car accident in 0.0461 seconds with 90.02 percentage AP. Moreover, the comparative experiments results show that YOLO-CA has comprehensive performance advantages of detecting car accident than other detection models, in terms of accuracy and real-time.

The comparative experiments are conducted for comparing seven detection models: (1) One-stage models: SSD, our proposed YOLO-CA, traditional YOLO-v3 and YOLO-v3 without MSFF (Multi-Scale Feature Fusion). (2) Two-stage models: Fast R-CNN, Faster R-CNN and Faster R-CNN with FPN. In order to comparatively demonstrate the validation of YOLO-CA as well as confirm its strength in terms of the comprehensive performance on the accuracy and real-time,

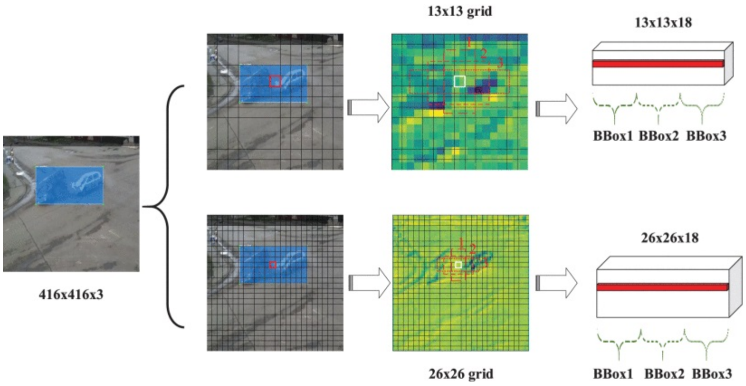


Figure 2.2: Architecture of proposed model

2.3 Car Crash Detection System using Machine Learning and Deep Learning Algorithms[3]

In conclusion this comparative analysis will lead to proper assessment of damage, guaranteed safety to victim life so that they can be saved quickly, lower energy consumption, better accuracy, more efficient use of time. The impact of a collision will determine the magnitude of damage; this data will allow us to determine the amount of damage caused that can be processed by running through insurance companies thereby getting a proper time to assess. Concluding that the CNN model will detect the confirmation of accident and severity and point of impact using video data set and using random forest and logistic regression to predict the patterns of how the accident are happening with severity and point of impact..

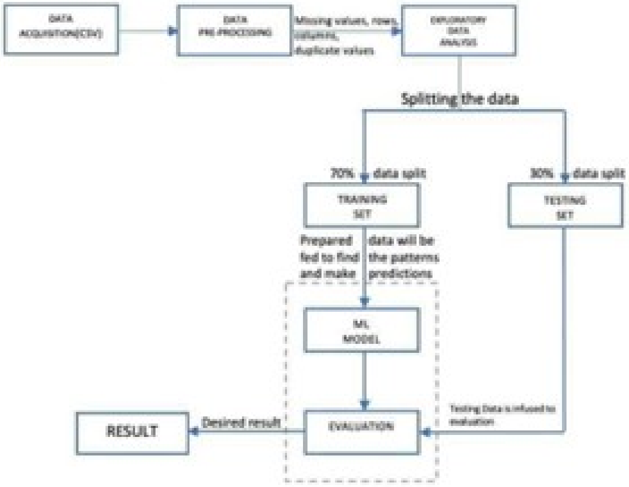

DATA ACQUISITION(CSV)
DATA PRE-PROCESSING
Missing values, rows, columns, duplicate values
EXPLORATORY DATA ANALYSIS
Splitting the data
70% data split
30% data split
TRAINING SET
TESTING SET
Prepared data will be fed to find the patterns and make predictions
ML MODEL
EVALUATION
Testing Data is infused to evaluation
Desired result
RESULT

Figure 2.3: Automatic Traffic Light

Usually for each lane 120 seconds of green light is on but in some areas of the city where traffic is less; in that intersection green light timing is less than 120 seconds. Totally it depends on traffic density in that area of the city. Before green light, yellow light glow for 20 seconds; indicating to start your vehicle and be ready to move. For all the time red light is on, indicating each vehicle to stop. This system cannot identify emergency vehicles like ambulance, VIP car etc. It treats all vehicles and emergency vehicles in the same way. Because they have fixed the timings for red and green signals and these signals are changing sequentially, but at the night time both red and green signals are manually switched off and only yellow signal will be switched on. So there is probability of delay in emergency services in peak hours. Therefore this technique is also inefficient in some times

CHAPTER 3

PROPOSED METHOD

3.1 Attention Fast R-CNN Overview

Figure 3.1 depicts the overview of our proposed Attention R-CNN. Attention R-CNN consists of two streams for two different tasks: the appearance stream and the characteristic stream. The appearance stream detects object appearance bounding boxes with their classes. Meanwhile, the characteristic stream utilizes the results from the appearance stream to recognize characteristic properties for all detected objects.

The accuracy percent of each of these algorithms when compared to each other in an image processing situation, it gives the values as...

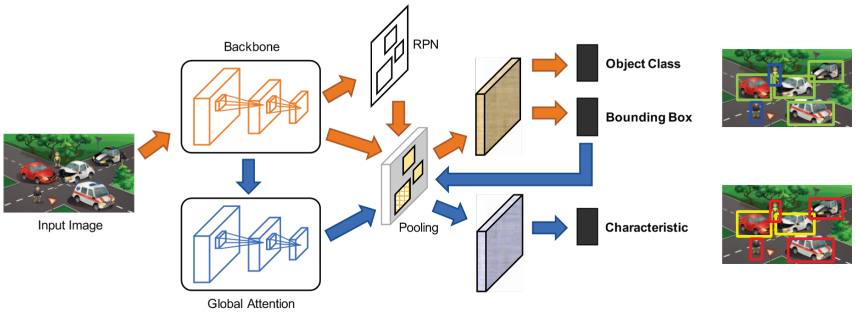


Figure 3.1: The overview of our proposed network. Attention FR-CNN consists of two streams, namely, the appearance stream (orange flow) for object class detection and the characteristic stream (blue flow) for object characteristic computation.

3.2 Appearance Stream

Faster R-CNN [30] is a state-of-the-art method presenting for two-stage detectors. Recent detectors [18], [25] usually follow the architecture of Faster R-CNN. However, the original Faster R-CNN is not effective in detecting road objects due to imbalance category and high density. Therefore, to efficiently detect objects on roads, we employ the appearance stream by adopting Faster R-CNN with modification. Particularly, we improve multi-scale features extracted from the backbone and RoI features exploited from the head network. We also solve imbalance of object categories in the training process through balance loss functions. Leveraging advantages of cutting-edge deep models, we use ResNet-50 [11] followed by Feature Pyramid Network (FPN) [18] as the backbone for the entire network. Following [25], we balance multi-level features by combining all features in the FPN with the average feature: Fi = Fi + Fave, where Fave = 1 N N i=1 Fi denotes the average feature. To integrate multi-level features and preserve their semantic hierarchy at the same time, we first resize the multilevel features to an intermediate size. The obtained features are then rescaled using the same but reverse procedure to strengthen the original features.

Following the standard design in two-stage detectors, we first detect the possible positions containing objects via Region Proposal Network (RPN) . RPN shares weights with the main backbone and outputs bounding boxes (RoI/object proposal) at various sizes. For each RoI, a fixedsize feature map (i.e., 7 × 7) is pooled from the image feature map using the RoIPool layer . We here replace the original RoIPool layer with a Precise RoI Pooling layer (PrRoI) because of its outstanding effect. The RoIPool works by dividing the RoI into a regular grid and then max-pooling the feature map values in each grid cell. This quantization, however, causes misalignment between the RoI and the extracted features due to the harsh rounding operations when mapping the RoI coordinates from the input image space to the image feature map space and when dividing the RoI into grid cells. On the other hand, PrRoI does not have the problem of misalignment. Indeed, PrRoI uses average pooling instead of max pooling for each bin and has a continuous gradient on bounding box coordinates. That is, one can take the derivatives of some loss function with respect to the coordinates of each RoI and optimize the RoI coordinates

After that, we extract features inside these proposals to compute object positions and object classes. Inspired by the head network architecture proposed by Lin et al. , RoI features are fed into a stack of four 3 × 3 × 256 convolution (conv) layer. Each conv layer is followed by a Group Normalization (GN) layer and a ReLU layer . Road object categories are obviously heavy imbalance. To address this problem, we employ balance loss functions. The balance loss of the appearance stream LbaApp is computed as follows:

$$\mathrm{L}baApp = \mathrm{L}obj + \mathrm{L}objLoc + \mathrm{L}cls + \mathrm{L}loc,$$

where Lobj and L$objLoc$ are the output of the RPN, Lcls and Lloc are defined on the output of the head network. The objectness loss is the Focal Loss computed as follows:

$$\mathrm{L}obj(\mathrm{p,u}) = -\alpha(1-pu)^{\gamma}.\mathrm{log}pu$$

$$\mathcal{L}_{\mathrm{cls}}(p,u) = -\frac{1-\beta}{1-\beta^{n_u}}\log p_u$$

where pu is the softmax output for the true class u, and nu is the number of samples in the ground-truth class u. Parameter $\beta = 0.999$ is computed similarly to [4]. The bounding box regression losses $L_{objloc}(tu, v)$ and $L_{loc}(tu, v)$ is computed as the Balance L1 Loss [25] between the regressed box offset t^u (corresponding to the ground-truth object class u) and the ground-truth box offset v:

$$\mathcal{L}_{\mathrm{BL1}}(t^u, v) = \sum_{i\in\{x,y,w,h\}} Balance_{\mathrm{L}_1}(t^u_i - v_i), \qquad (4)$$

where

$$Balance_{\mathrm{L}_1}(x) = \begin{cases} \frac{\alpha}{b}(b|x|+1)\ln(b|x|+1) & \text{if } |x| \\ \gamma|x| + C & \end{cases}$$

where $\alpha ln(b+1) = \gamma$. Parameters $\alpha = 0.5$,$\gamma = 1.5$ are computed similarly

3.3 Characteristic Stream

The characteristic stream consists of two modules: Global Attention and head network as shown in Fig. 1. The Global Attention converts features extracted from the backbone of the appearance stream to characteristic features, which is then useful for object characteristic property computation. The head network outputs the probability of characteristic categories for all the objects detected by the appearance stream.

1) Global Attention: Features extracted from the appearance stream are not effective in object characteristic property computation. This is because appearance features do not necessarily reflect implicit properties of objects (even though these features can recognize car/motor, they may not be able to distinguish safety/damage). To address this problem, we build on top of the backbone the Global Attention module to convert appearance features to effective characteristic features. The Global Attention involves three components: transformer, attention map generator, and attention mechanism.

Transformer: It converts appearance features to characteristic features. We first extract an average feature from multi-level features of the backbone. We note that multi-level features are resized to the same size before the extraction. The average feature is then fed into a stack of four 3×3×256 conv layers. Each conv layer is followed by a GN layer [36] and a ReLU

Attention map generator: This is developed by employing the Attention Branch Network (ABN) [8] with modifications. The customized ABN contains an attention branch and a perception branch. The attention branch consists of a block of five conv layers. The first four conv layers has 3×3×256 kernel. Each conv layer is followed by a GN layer [36] and a ReLU . The last conv layer is 1 × 1 × K, where K is the number of classes. After that, we use a 1 × 1 × K conv layer for pixel-wise classification, followed by a Global Average Pooling (GAP) layer [17] to generate a probability vector of classes. In order to aggregate K feature maps, a 1 × 1 × 1 conv layer is plugged-in right after the conv block as the attention generator. Fig. 2 illustrates examples of the generated attention map. On the other hand, the perception branch outputs the probability of each class, following the standard design of classification models. Similarly to [8], the generated attention map is applied to the transformed feature by an attention mechanism to enhance feature maps: $\mathrm{G} = (1 + \mathrm{M}) \cdot \mathrm{Ftf}$, where M is the attention map, Ftf is the transformed feature, and G is the new feature. A Max Pooling layer is used to obtain a fixed-size 64 × 64 feature. The feature is then fed into two fully connected layers, yielding feature vectors with

1024 and K channels, respectively

The loss of Global Attention is also the loss of ABN, computed as follows:

$$L_{\text{gbAtt}} = L_{\text{att}} + L_{\text{per}}$$

where L_{att} denotes the training loss at the attention branch, and L_{per} denotes the training loss at the perception branch. All the two losses are the Class-Balance Softmax CrossEntropy Loss [4] computed as follows:

$$\mathcal{L}_{\text{cls}}(p, u) = -\frac{1-\beta}{1-\beta^{n_u}} \log p_u$$

where Pu is the softmax output for the true class u, and Nu is the number of samples in the ground-truth class u. Parameter $\beta = 0.999$ is computed similarly .

Attention mechanism: It aims to output characteristic features from multi-level appearance features F_{i} by combining them with the transformed feature F_{tf} and the attention map M (see block Attention A2 in Fig. 2) as follows:

$$F\tilde{}_{\text{i}} = (1 + \text{M}) \cdot (F_{\text{tf}} + F_{\text{i}}),$$

where $F\tilde{}_{\text{i}}$ is multi-level characteristic features, corresponding to appearance features F_{i}. In practice, we directly apply the formula below to utilize the pre-computed feature g:

$$F\tilde{}_{\text{i}} = (1 + \text{M}) \cdot F_{\text{i}} + \text{G}.$$

2) Head Network: To compute the characteristic properties of detected objects, for each target, we employ the head network to exploit RoI features. For each target RoI, RoI feature is exploited directly from the region inside the target RoI through a PrRoI layer [12]. Similarly to the appearance stream, the RoI features F$\tilde{}$ are first fed into a stack of four 3 × 3 × 256 conv layers, in which each of them is followed by a GN layer [36] and a ReLU . They are then fed into two fully connected layers, yielding feature vectors with 1024 and K channels, respectively.

The loss of characteristic classifier is the Class-Balance Softmax Cross-Entropy Loss [4] computed as follows

$$L_{\text{chaCls}}(p, u) = -\frac{1-\beta}{1-\beta^{n_u}} \log p_u$$

where p_{u} is the softmax output for the true class u, and nu is the number of samples in the ground-truth class u. Parameter $\beta = 0.999$ is computed similarly to. The balance loss of the characteristic stream L_{baCha} is computed as follows:

$$L_{\text{baCha}} = L_{\text{gbAtt}} + L_{\text{chaCls}}$$

3.4 ACCIDENT DETECTION VIDEO DATASET

To promote researches on accident detection, a publicly available dataset is mandatory. We thus construct an Accident Detection Video (ADV) dataset. We emphasize that no other dataset is publicly available for accident detection. To build the new dataset, we selected 100 raw videos recorded by Chan et al. [1], each of which involves at least a dangerous/crashed event, and annotated ground-truth for accident detection. We initially annotated a bounding box with its object label for all road objects at every video frame where we used the semi-supervised annotation process [16]. We used 7 popular semantic labels to annotate road object classes. They are car, bus, truck, motorbike, bicycle, pedestrian, and rider. After that, we manually labeled three accident categories for all objects: safe, dangerous, and crashed.

Our newly constructed ADV dataset is the first dataset designed explicitly for the task of accident detection. The dataset consists of 100 videos, each of which has 100 frames, with total 88,800 object bounding box ground-truth. Some examples are shown in Fig. 3 with the corresponding groundtruth label annotations.

The main difference of our ADV dataset from existing road object datasets [3], [23], [39] having a dominant numberof four-wheel vehicles is that our ADV dataset consists of a large number of persons and two-wheel vehicles. This reflects the specialty of many Asian countries (cf. Table I). The ratios of each category are shown in Fig. 4. We remark that we count only images having object bounding box groundtruth because existing datasets do not have groundtruth for all images. In the ADV dataset, it is noteworthy that when accident events happen, strange poses of persons and vehicles appear (cf. Fig. 5), which never existed in other datasets [3], [23], [39]. This makes the detection more challenging than existing datasets. The ADV dataset also contains videos in different weather (sunny, rainy, snowy) and timeline (daytime and night) to increase the difficulty of the detection.

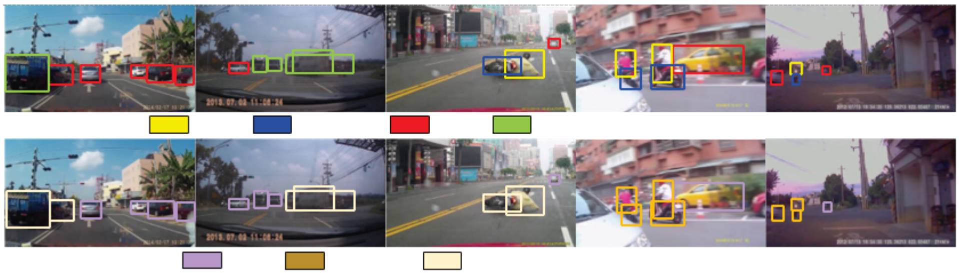
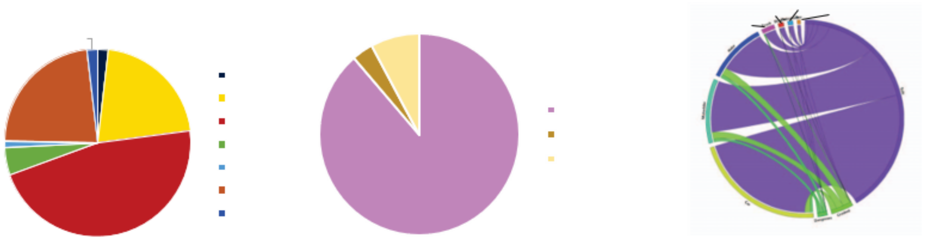


Figure 3.2: Examples of Accident Detection Video (ADV) dataset. The first row is object class ground-truth and the second row is accident ground-truth. We do not highlight tiny objects and unimportant objects (blurred, heavily occluded).

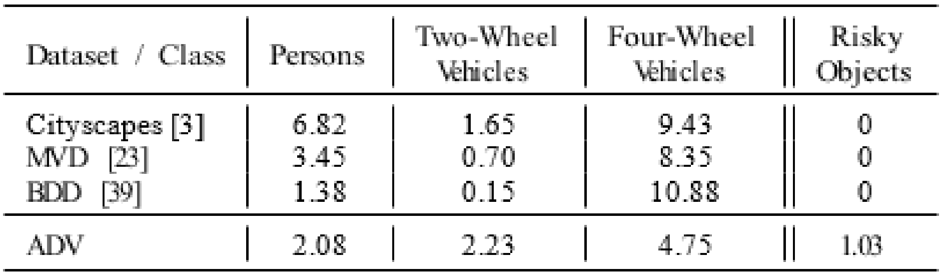


(a) Object categories.

(b) Accident categories.

(c) Object-Accident distribution.

Figure 3.3: Categories distribution over the ADV dataset.



Dataset / Class	Persons	Two-Wheel Vehicles	Four-Wheel Vehicles	Risky Objects
Cityscapes [3]	6.82	1.65	9.43	0
MVD [23]	3.45	0.70	8.35	0
BDD [39]	1.38	0.15	10.88	0
ADV	2.08	2.23	4.75	1.03

Figure 3.4: Number of objects at each image over different datasets (%). Risky objects indicate dangerous and crashed vehicles and persons

CHAPTER 4

EXPERIMENTS

4.1 Benchmark Dataset and Evaluation Criteria

Our constructed ADV dataset is randomly divided in to the trainval set and the test set with ratios 80% and 20%. All reported results follow standard COCO-style Average Precision (AP) metrics that include AP (averaged over IoU thresholds), AP50 (AP for IoU threshold 50%), AP75 (AP for IoU threshold 75%) [20]. In addition, mean Average Precision (mAP) denotes the AP over all the categories.

4.2 Implementation Details

For fair comparisons, all experiments are implemented on PyTorch, and based on the published code of Mask R-CNN Benchmark [22] .

We trained detectors with two GTX 1080Ti GPUs (4 images per GPU) for 20 epochs. We used the Stochastic Gradient Descent (SGD) optimization with a moment $= 0.9$ and a weight decay of 0.0001. The training process was conducted by initially fine-tuning the available pretrained model from the MS-COCO dataset [20] on the BDD dataset [39] to transfer domain from general objects to road objects. After that, we fine-tuned models on the newly constructed ADV dataset.

It is not easy to directly train the whole network in an end-to-end manner; different learning rates are employed for different streams. Here, we employ a two-stage optimization approach to solve this problem. We first train the appearance stream with an initial learning rate of 0.02, using the loss LbaApp. We then train the characteristic stream with an initial learning rate of 0.002, using the loss LbaCha. The learning rate decrease by 0.1 after 8, 11, and 16 epochs, respectively

CHAPTER 5

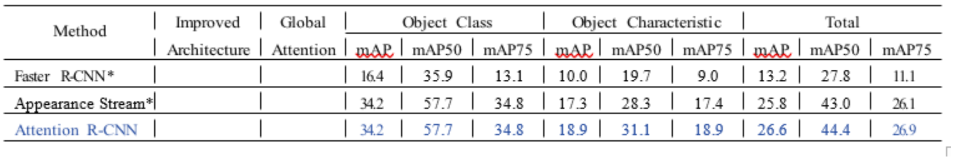
RESULT

We emphasize that Attention R-CNN is the first work for the task of accident detection, meaning that no state-of-the-art method is available for comparison. In this section, we thus compare our proposed Attention R-CNN with baselines. We investigate the impact of different components in our proposed Attention R-CNN, such as Global Attention and improved network architecture of the appearance stream, including balance losses. Experimental results are shown in Table II. Figure 6 shows the visual comparison of different meth-ods. As illustrated in the figure, our Attention R-CNN yield better results than Faster R-CNN. Our results are close to the ground truth and focus on risky objects (e.g., dangerous and crashed vehicles and persons). As shown in Table II, our proposed Attention R-CNN significantly outperforms all baselines

Faster R-CNN*: Faster R-CNN [18] is designed only for object class and bounding box detection. To additionally compute object characteristics, we attach more two fully connected layers, followed by a classifier after the RoI pooling layer for object characteristic recognition, denoted by Faster R-CNN*. Table II shows that our Attention R-CNN significantly outperforms Faster R-CNN*.

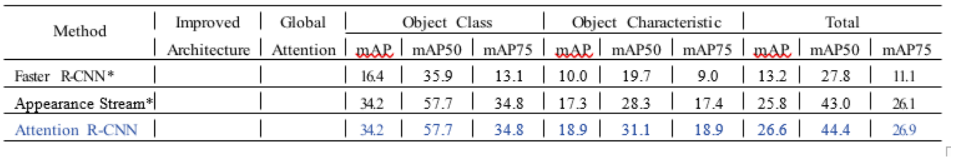
Global Attention: We investigate the performance of the Global Attention by comparing our completed Attention R-CNN with the one without Global Attention, denoted by Appearance Stream*. This baseline is implemented similarly to our appearance stream with an additional head network for object characteristic detection. As shown in Table II, our completed Attention R-CNN surpasses Appearance Stream*, highlighting the contribution of our Global Attention.

Improved network architecture: We suspect that one of the major factors affecting the accident detection was the network architecture (including the backbone, the head network, and loss functions). As also seen in Table II, all networks with the improved network architecture (i.e., Appearance Stream* and Attention R-CNN) significantly outperforms Faster R-CNN* based on the old designed architecture. This clearly shows the importance of the new network architecture and balance losses in the proposed method



Method	Improved	Global	Object Class			Object Characteristic			Total		
	Architecture	Attention	mAP	mAP50	mAP75	mAP	mAP50	mAP75	mAP	mAP50	mAP75
Faster R-CNN*			16.4	35.9	13.1	10.0	19.7	9.0	13.2	27.8	11.1
Appearance Stream*			34.2	57.7	34.8	17.3	28.3	17.4	25.8	43.0	26.1
Attention R-CNN			34.2	57.7	34.8	18.9	31.1	18.9	26.6	44.4	26.9

Figure 5.1: Experimental results on ADV dataset. Methods are evaluated by mAP, which can justify both bounding box and recognition of each detection.



Method	Improved	Global	Object Class			Object Characteristic			Total		
	Architecture	Attention	mAP	mAP50	mAP75	mAP	mAP50	mAP75	mAP	mAP50	mAP75
Faster R-CNN*			16.4	35.9	13.1	10.0	19.7	9.0	13.2	27.8	11.1
Appearance Stream*			34.2	57.7	34.8	17.3	28.3	17.4	25.8	43.0	26.1
Attention R-CNN			34.2	57.7	34.8	18.9	31.1	18.9	26.6	44.4	26.9

Figure 5.2: Detection results of Attention R-CNN in each category (mAP50) : Detection results of Attention R-CNN at different conditions

5.1 Analysis and Discussion

We show results of Attention R-CNN in different conditional environments in Fig. 5.1. Our proposed method can achieve good performance in both daytime and nighttime, which have differences in light intensity distribution. Results of Attention R-CNN in each category are shown in Figure 5.2. mAP scores of Bus and Dangerous are too low, compared with other categories. Due to the imbalance of the ADV dataset (cf. Fig.5.1), bus takes only 1% of objects in the ADV dataset, leading to miss-detect to car or truck In dangerous events, almost vehicles change only their behaviors but not appearance. Meanwhile, the proposed Attention R-CNN is frame-by-frame processing, which lacks temporal information to detect behavior changes. In the future, we will investigate effectiveness of temporal information to improve the performance of the system

CHAPTER 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

We addressed a new task of accident detection, and provided a dataset for this task. We gave the baseline for the task on the provided dataset. We believe that our dataset will promote new advancements in accident detection. Our proposed Attention R-CNN network for accident detection consists of two streams: object detection stream and object characteristic stream. The object characteristic stream employs the attention mechanism that exploits local and global contextual-levels of a detected object using not only its corresponding region but also entire of the scene to recognize the object characteristic property. This leads to significant improvement in both object class detection and object characteristic detection, establishing a baseline on our provided dataset. Besides extending the quantity of the dataset, developing a way to exploit temporal information from videos is left for future work.

6.2 FUTURE SCOPE

Improved accuracy and robustness: With more data and better algorithms, it is possible to develop collision detection systems that are more accurate and robust than current systems. This would lead to fewer false positives and false negatives, making the systems more reliable and trustworthy. Real-time detection: Current collision detection systems typically operate on pre-recorded video footage. However, with the advent of powerful edge computing devices, it is now possible to develop collision detection systems that can operate in real time. This would allow the systems to be used in a variety of new applications, such as autonomous vehicles and safety monitoring systems. Multi-sensor fusion: Collision detection systems can be made more accurate and robust by fusing data from multiple sensors, such as cameras, radar, and lidar. This would allow the systems to detect objects in a wider range of conditions and environments. Deep learning: Deep learning is a powerful machine learning technique that is being used to achieve state-of-the-art results in a variety of computer vision tasks, including collision detection. As deep learning algorithms continue to improve, we can expect to see collision detection systems that are even more accurate and robust than current systems.

REFERENCES

[1] S. Ghosh, S. J. Sunny and R. Roney, "Accident Detection Using Convolutional Neural Networks," 2019 International Conference on Data Science and Communication (IconDSC), Bangalore, India, 2019, pp. 1-6, doi: 10.1109/IconDSC.2019.8816881.

[2] D. Tian, C. Zhang, X. Duan and X. Wang, "An Automatic Car Accident Detection Method Based on Cooperative Vehicle Infrastructure Systems," in IEEE Access, vol. 7, pp. 127453-127463, 2019, doi: 10.1109/ACCESS.2019.2939532.

[3] Saravanarajan, V.S., Chen, RC., Dewi, C. et al. Car crash detection using ensemble deep learning. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-15906-9

[4] L. Taccari, F. Sambo, L. Bravi, S. Salti, L. Sarti, M. Simoncini, and A. Lori. Classification of crash and near-crash events from dashcam videos and telematics. In International Conference on Intelligent Transportation Systems, pages 2460–2465, Nov 2018.

[5] F. Yu, W. Xian, Y. Chen, F. Liu, M. Liao, V. Madhavan, and T. Darrell. Bdd100k: A diverse driving video database with scalable annotation tooling. arXiv:1805.04687, 2018.

[6] F.-H. Chan, Y.-T. Chen, Y. Xiang, and M. Sun. Anticipating accidents in dashcam videos. In ACCV, pages 136–153, 2016.

[7] Saravanarajan, V.S., Chen, RC., Dewi, C. et al. Car crash detection using ensemble deep learning. Multimed Tools Appl (2023). https://doi.org/10.1007/s11042-023-15906-9

[8] K.-T. Nguyen, T.-H. Hoang, M.-T. Tran, T.-N. Le, N.-M. Bui, T.-L. Do, V.-K. Vo-Ho, Q.-A. Luong, M.-K. Tran, T.-A. Nguyen, T.-D. Truong, V.-T. Nguyen, and M. N. Do. Vehicle re-identification with learned representation and spatial verification and abnormality detection with multi-adaptive vehicle detectors for traffic video analysis. In CVPR Workshops, June 2019

[9] T.-N. Le, A. Sugimoto, S. Ono, and H. Kawasaki. Toward interactive self-annotation for video object bounding box: Recurrent self-learning and hierarchical annotation based framework. In IEEE Winter Conference on Applications of Computer Vision, 2020.

[10] X. Zhang and X. Zhu, "Vehicle Detection in the Aerial Infrared Images via an Improved Yolov3 Network," 2019 IEEE 4th International Conference on Signal and Image Processing (ICSIP), 2019, pp. 372-376, doi: 10.1109/SIPROCESS.2019.8868430.

[11] Z. B. Jiang, R. Luo, J. Mao, T. Xiao, and Y. Jiang. Acquisition of localization confidence for accurate object detection. In ECCV, pages 784–799, 2018.