A Novel Approach for Road Accident Detection using DETR Algorithm

Aparajith Srinivasan, Anirudh Srikanth, Haresh Indrajit, Venkateswaran Narasimhan
Department of Electronics and Communication Engineering
SSN College of Engineering
Kalavakkam, India
Email: aparajith17022@ece.ssn.edu.in, anirudh17017@ece.ssn.edu.in, haresh17049@ece.ssn.edu.in, venkateswarann@ssn.edu.in

***Abstract*—Road accidents are man-made cataclysmic phenomena and are not generally predictable. With increasing numbers of deaths due to accidents in the roadways, a smart and fast detection system for road accidents is the need of the hour. Often, precious few seconds after the accidents make the difference between life and death. To address this problem more efficiently, "A Novel Approach for Road Accident Detection in CCTV Videos using DETR Algorithm" has been developed to aid in notifying hospitals and the local police at places where instant notification is seldom feasible. This paper presents a novel and efficient method for detecting road accidents with DETR (Detection Transformers) and Random Forest Classifier. Objects such as cars, bikes, people, etc. in the CCTV footage are detected using the DETR and the features are fed to a Random Forest Classifier for frame wise classification. Each frame of the video is classified as an accident frame or a non-accident frame. A total count of predicted accident frames from any 60 continuous frames of the video are considered using a sliding window technique before the final decision is made. Simulation results show that the proposed system achieves 78.2% detection rate in CCTV videos.**

Keywords—Road Accident Detection, CCTV Videos, Detection Transformer, Random Forest Classifier, Sliding Window Technique

I. INTRODUCTION

Transportation has become a necessity in today's world not only for the movement of people, but also for the movement of resources which have a tremendous impact on a country's economy. Among all the different means of transportation, roadways have become the most common way due its cost effectiveness without a compromise in speed. But it has also become a reason for concern due to the massive number of accident related deaths.

Across the world, almost 1.35 million people lose their lives every year due to road accidents [1]. More than 90% of these deaths occur in developing or underdeveloped countries. In addition to this, road accidents have become the major cause of deaths of children and young adults aged 5 to 29 years. In India, the automobile population growth is much more compared to the growth of the economy and human population. It is estimated that every hour, approximately 17 road accident mortalities occur in India [2]. Injuries due to road accidents become more fatal when the time taken for providing immediate attention for the injured increases. Attending to the injuries after the accident is time sensitive: delay in minutes is the deciding factor between life and death. Moreover, an equal percentage of people live with permanent disability on account of delayed post-crash care and poor crisis management.

It is practically impossible for the government / NGO / law enforcers to monitor all the highways 24/7 as it requires huge manpower. Majority of the highway accidents occur between midnight to 4am and between 3pm to 6pm [3]. In addition to this, accidents that occur in highways do not get the timely attention as highways usually tend to be deserted at most of the times.

This work proposes a novel model to detect road accidents using CCTV cameras with the following features:

- Incorporation of a recently proposed object detection algorithm called the DETR (Detection Transformer) which employs a simpler architecture compared to the other object detection algorithms.
- Detection based on correlation between all the objects present in the image.
- Improved speed and accuracy compared to already existing work.

II. RELATED WORK

A. Object Detection

Object detection research has been growing more and more robust over the past decade. From the HOG (Histogram of Oriented Gradients) based object detection [4] until the recent state-of-the-art single stage object detectors [5], the computer vision community has witnessed various design iterations in the object detection pipeline. The HOG based object detection pipeline proposed by [6] uses a lot of heuristics until the final detection is obtained. Some of the heuristics used by them are computation of gradients, weighting of votes into spatial and orientational cells, histogram normalization, collection of HOG features over the detection window, followed by a linear SVM (Support Vector Machine) to get the final detection.

Once the effectiveness of deep learning based algorithms was realized by the research community, leveraging of features computed by DNN (Deep Neural Networks) started to increase [7]. The R-CNN (Region based Convolutional Neural Network) used in [28], has utilized a region proposal algorithm which is called the Selective Search Algorithm. Following this, the proposed regions were fed to a CNN for feature extraction. The features calculated are sent to an SVM trained to classify each class. The greedy non-maxima suppression is applied during test-time detection to avoid duplicate detections. This pipeline served as a base template for the computer vision community. Therefore, improvements were made in works such as Fast R-CNN [8], which fed the input image to a CNN obtaining a resultant feature map, and crops from the resulting feature map was obtained to find candidate objects containing regions. ROI (Region of Interest) Pooling was applied to those candidate crops to make them compatible as inputs for the fully connected networks present downstream. The Faster R-CNN [9], overcame the bottleneck of region proposal generation in

978-1-7281-8376-3/20/$31.00 ©2020 IEEE

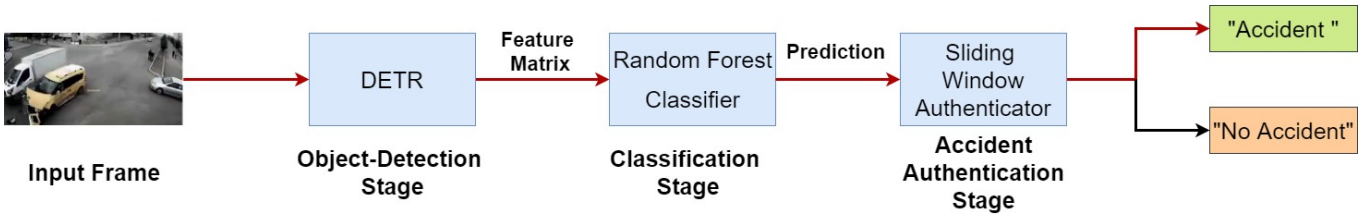

Fig. 1. Proposed Approach

Fast-RCNN by using a region proposal network to generate candidate regions, which resulted in more inference speed. The onset of one-stage detectors like YOLO (You Only Look Once) [10] and SSD (Single Shot Multibox Detectors) [11] were seminal for the development of architectures that can be trained end-to-end on object detection tasks instead of having to train separate networks that will be a part of the final network like those in Fast R-CNN and Faster R-CNN.

However, even the state-of-the-art one-stage detectors like YOLOv3 [12] require the encoding of domain information in the form of anchor boxes prior to training. This requires a lot of heuristics which makes the handling of one-stage detectors to be done meticulously. Moreover, the need for the greedy non-maxima suppression in YOLO and its variants, makes it slightly compute-expensive.

All these bottlenecks were overcome by the introduction of the DETR (Detection Transformer) by Facebook AI research [13]. The advantages of transformers such as its multi-headed attention mechanism and positional encodings are leveraged in DETR. There is no need for the encoding of prior domain information in the form of anchor boxes unlike YOLO or SSD. In addition to this, DETR has an architecture which is easy to comprehend and also flexible.

B. *Road Traffic Anomaly Detection*

State-of-the-art computer vision techniques have been leveraged in the road traffic anomaly detection a lot in recent times [14]. There are various approaches to anomaly detection in road traffic namely, model-based, classification-based, proximity-based, prediction-based and reconstruction-based approaches [14]. Past works have used SVM in a binary classification problem where the SVM demarcates the anomalous and normal classes [15]. However, the features used by the SVM-based anomaly detection models are mostly HOG features [16]. Calculation of HOG features and then feeding it to the anomaly detection model as inputs can increase latency and also be compute-intensive.

DNN (Deep Neural Network) based architectures extract high-level features automatically without explicit feature engineering in [14]. There has been extensive use of DNN based object detection algorithms for road traffic surveillance. In [17], YOLOv3[12] has been used for feature extraction followed by a decision tree algorithm used downstream for the classification task. Even though YOLOv3 had a very fast inference, it performed less accurately in dense traffic conditions and failed to detect fast moving vehicles.

In [19], Mask R-CNN [18] was used for the initial object detection stage. Mask R-CNN performs poorly in scenarios involving very small objects (pedestrians and smaller vehicles). In addition to this, the rule-based approach incorporating the centroid tracking algorithm for accident detection has a pipeline with a lot of heuristics which may slow down the accident detection process.

In [20], autoencoders are used to localize the accident regions instead of the conventional object detection approach. The autoencoders tries to model the motion and appearance features of the frames using a video volume as an input. However, this approach is computationally intensive.

Following these footsteps, the presented approach uses the DETR as the candidate for object detection. A distinguishable feature of DETR is that it has a comparably similar number of parameters as Faster-RCNN, but has a faster interference. The features obtained from the DETR are used for accident detection. This is done by incorporating the Random Forest Classifier Algorithm. In addition to these two stages, this work includes an accident authentication stage which reduces the rate of false alarms.

III. Proposed Approach

Broadly, there are three stages in the proposed framework. The first stage is object detection which uses DETR to detect objects pertaining to road accidents such as motorcycles, cars, trucks, buses and persons. The features obtained are fed to the next stage. The second stage is accident classification which constitutes the Random Forest Classification algorithm [21] to predict the occurrence of an accident in the current frame. The third stage is the accident authentication stage executed using a sliding window technique which confirms the occurrence of an accident based on the predictions from the previous frames.

A. *Object Detection Stage*

The DETR has a CNN (Convolutional Neural Network) backbone, a transformer encoder-decoder block and fully connected layers for the class and bounding box predictions. The image is input to the backbone CNN which is a ResNet [22]. The obtained feature map from the last convolutional layers of the ResNet is down sampled using a reduction layer. The down sampled features are fed to the transformer encoder - decoder block. The transformer decoder outputs a 100×256 vector, out of which the most influential features are selected, resulting in a 10×256 feature vector used in the downstream classification algorithm.

1) ResNet: The ResNet architecture [22] is based on the fact that learning perturbations from the identity mapping is more computationally efficient than learning a function mapping from scratch. For an input x, if the desired mapping is $H(x)$ when fed through a set of parameterized layers, the layers are forced to learn a residual mapping $F(x)$ such that $F(x) = H(x) - x$. It was found that learning this residual

mapping *F(x)* was easier for the stacked non-linear layers than learning $H(x)$ from scratch. Thus, the ResNet architecture was seminal in dealing with the problem of vanishing gradients, exploding gradients and the degradation problem in very deep neural network architectures.

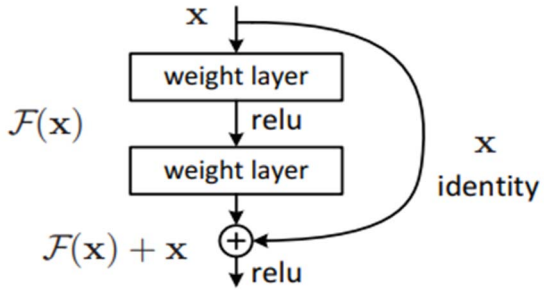

Fig. 2. Residual Learning Block [22]

In the DETR, a 50-layer ResNet pre-trained on the ImageNet dataset [30] was used. It was imported without the classification heads, since the final convolutional layer's feature map representation of the input image was only necessary.

2) Transformer: The output feature map from the ResNet's convolutional layers are of the shape $H \times W \times C$ where $H = H_0/32$, $W = W_0/32$ and $C = 2048$. Here H_0 and W_0 are the height and width of the input image and C is the number of channels in the ResNet feature map. The depth of the feature map is reduced to d, where $d = 256$. The first sub-encoder in the encoder block of the transformer expects a sequence as an input. So the $H \times W \times d$ feature map is reshaped into a vector of length $d \times (H \times W)$ and is fed to the encoder block.

The feature vector, $X \in \mathbb{R}^{d \times (H \times W)}$ will be the input to the encoder as shown in Fig. 3. W^Q, W^K and W^V are the query, key and value weight matrices that are randomly initialized and are optimized through the training process. The operations in equations (1) , (2) and (3) are done to obtain the Query matrix Q, Key matrix K and Value matrix V. There will be d rows in each of K, Q and V where each row can be intuitively understood to be a feature vector with $H \times W$ features.

$$Q = X * W^Q \tag{1}$$

$$K = X * W^K \tag{2}$$

$$V = X * W^V \tag{3}$$

Where $*$ represents the matrix operation.

The encoder is permutation-invariant [23]. However, it is desirable to preserve the semantics of the image. To ensure this, spatial positional encodings [24] are added to the K and Q matrices before inputting them to the encoder block as seen in Fig. 3. The multi-head self-attention block shown in Fig. 3. is responsible for encoding the input image features X such that each feature vector $x_i \in \mathbb{R}^{H \times W}$ is encoded by keeping focus on other feature vectors $x_j \in \mathbb{R}^{H \times W}$. This is achieved by the scores obtained in the following equation.

$$O = softmax\left(\frac{Q * K^T}{\sqrt{d}}\right) * V \tag{4}$$

The dimensions of O are same as that that of K , Q and V. Each row of O represents a vector of size $H \times W$ in which each element O_{ij} (where $i \in [0, d]$ and $j \in [0, H \times W]$) represents a score which numerically conveys how much attention should be kept on a feature cell with respect to the whole image feature map.

The multi-head attention block consists of eight heads. Each head has its own W^Q, W^K and W^V matrices. This has the advantage of producing a different representation subspace with each attention head, resulting in an encoding of the image features with respect to a global context rather than a local neighborhood. Each attention head's output O is concatenated along the columns to produce a matrix O_{concat}. This matrix is transformed further by equation (5) and is input to the fully connected layer with shared weights.

$$\hat{O} = O_{concat} * W^O \tag{5}$$

Where , W^O is a learnt weights matrix.

$\hat{O}$ is output of the first encoder of the encoder block. This will be the input of the subsequent encoder of the block. The number of encoders and decoders in each block is six. However, it is a hyperparameter that can be tuned as per one's discretion. Empirically, six encoders and decoders have produced the best results [13].

The architecture of the transformer decoder is similar to the encoder, with minor differences. The difference comes in the addition of object queries being the input to the first decoder layer of the decoder block and the presence of an encoder-decoder attention block. Each of the object queries are randomly initialized vectors where each object query is responsible for localizing on different parts of the image. The number of such object queries can be decided from the maximum number of objects of interest present in the training set images. In [13], the number of object queries were set to 100. The output of the entire encoder block is fed to the encoder-decoder attention block of the decoder, to which it serves as the input Value and Key matrices. The encoder-decoder attention block helps in giving attention to the image features which are vital for a particular detection. The spatial positional encodings are added to those Key and Query matrices. In both the encoder and decoder blocks, there are residual connections, where the input of an encoder or decoder layer is added with the output of the attention block of that layer. After this, the added result is layer normalized [25]. The final output of the decoder block is in turn fed to two fully connected layers, one to output the class prediction and the other to regress the bounding boxes. The class prediction and the bounding box forms a set prediction made by the DETR.

Unlike Recurrent Neural Networks [31] which require features to be fed one after the other, by utilizing transformers in DETR, image features are fed at one go. This has reduced latency in producing results. The parallel computing feature in the multi-head self-attention block and the shared weights in the fully connected layers help in making DETR, a detection algorithm which performs on par with modern object detectors in terms of accuracy and speed. Hence, this paper has incorporated DETR for the purpose of vehicular collision detection.

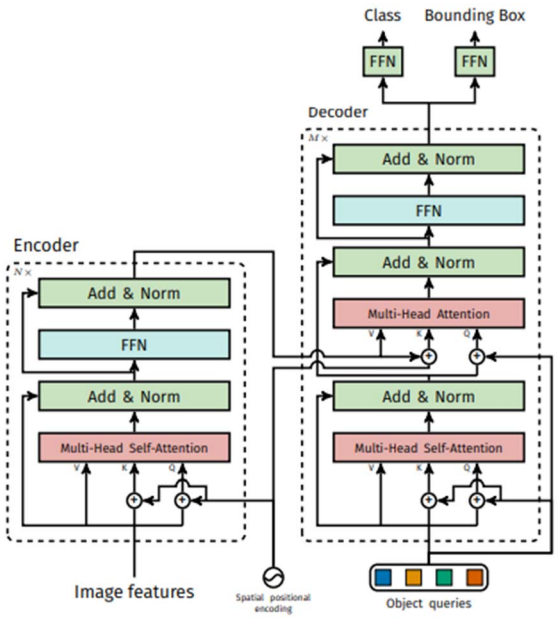

Fig. 3. Transformer encoder-decoder architechture used in DETR [13]

3) Loss Function: For N object queries and y being the set of ground truth values, there are N sets of predictions, $y' = \{y_i'\}_{i=1}^{N}$. For this, a bipartite matching loss is defined such as

$$\acute{\alpha} = \arg\min_{\alpha \in \varphi_N} \sum_{i=1}^{N} (L_{match}(y_i, y_{\alpha(i)}) \quad (6)$$

Equation (6) finds the permutation α of y' where $\alpha \in \varphi_N$ and φ_N is the set of all possible permutations of y. The equation finds the permutation of the set predictions that corresponds to the least matching cost $L_{match}(y_i, y'_{\alpha(i)})$, which is a pair wise matching cost between ground truth set y and predicted set y'.

B. Accident Classification Stage

The DETR outputs a 100×256 feature matrix for each frame of the video out of which only the features pertaining to the car, bus, motorcycle, truck and such classes which can be possible participants of a road accident are selected. After this processing, each feature vector from the feature matrix assumes the shape 1×2560. To classify each frame to be "accident frames" or "non-accident frames", this feature vector is fed to the Random Forest classifier [21].

1) Entropy: Of the two metrics, Gini coefficient and Entropy, the latter has been used in the presented work. Entropy was found to produce better results for the dataset [26], to decide which split on a node in the constituent decision trees in a random forest is better. Entropy is given by

$$Entropy = \sum_{i=1}^{N} (-p_i \times log(p_i)) \quad (8)$$

The number of constituent decision trees were chosen to be 500 and the maximum depth of the trees to be 40, as they empirically gave the best results.

C. Accident Authentication Stage

The sliding window technique has been used to authenticate the occurrence of an accident and to reduce the errors due to false alarms. A window in the form of a queue of size 60 is used which takes into account the predictions from past 60 frames. At start, the window is initialized with zeros. When a prediction has been made, the window dequeues the last element and enqueues the newly predicted value (either a one or zero) into the beginning of the window. Hence, at each instant, the window considers the latest 60 predictions. An accident is confirmed when the number of 'ones' present inside the sliding window is more than 50.

IV. Results and Discussions

A. Dataset Used

The most comprehensive dataset on accident footages was found to be the CADP: A Novel Dataset for CCTV Traffic Camera based Accident Analysis [26]. It contains 1416 accident footages at various timings of the day and weather conditions from the countries around the world. Thus, we found it most suitable to train and test our vehicular collision detection framework on this dataset. From these videos, a total of 7000 frames in which about 3500 accident frames and 3500 non-accident frames were used for training the model.

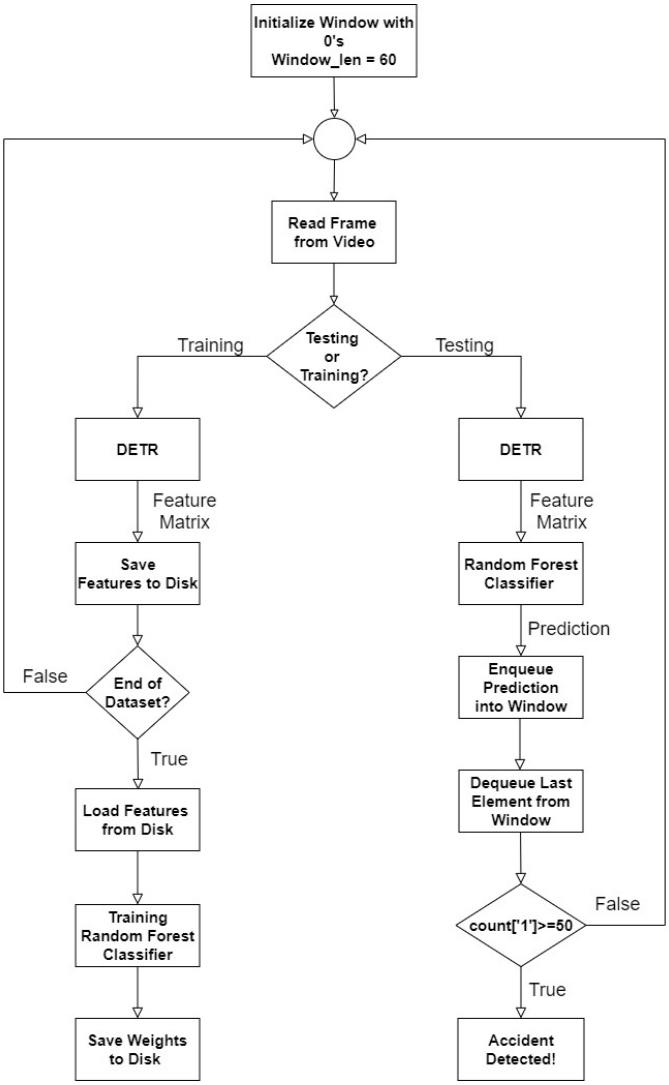

Fig. 4. Flowchart of the framework

TABLE I. PERFORMANCE METRICS

Metrics	Accuracy (%)	Precision '1'	F1 Score '1'	Recall '1'	AUC
Values	78.7	0.77	0.77	0.78	0.82

B. *Hardware and Software Dependencies*

All experiments were performed in Google Colab with a dual core Intel Xeon processor @2.3 GHz and 13 GB RAM. It is equipped with a NVIDIA Tesla K80 (GK210 chipset), 12 GB RAM, 2496 CUDA cores @560 MHz. PyTorch 1.5.1 was used for the object detection stage. OpenCV 4.1.2 was used for the image processing tasks. Scikit-learn 0.22.2 was used for classification.

C. *Simulation Results*

The presented framework was trained on over the 7000 frames from the dataset. The confusion matrix for the result obtained post-training and testing the model with 718 different CCTV frames is shown below.

TABLE II. CONFUSION MATRIX

Prediction/Ground Truth	Accident Frames '1'	Non-Accident Frames '0'
Predicted Accident '1'	266	79
Predicted No Accident '0'	74	299

Table I shows the common performance metrics used to evaluate the model. Following [27], the DR (Detection Rate) has been used as an evaluation metric for the presented framework. It is numerically equivalent to the value of recall percentage.

$$DR = \frac{Detected\ accident\ frames}{Total\ Accident\ Frames} \times 100\ \% \qquad (9)$$

For the presented work, the false positive and false negative predictions are comparatively lesser than the true predictions. This can be inferred from the confusion matrix and also the significantly high values of precision and recall. The ROC curve for the classifier is given in Fig. 5.

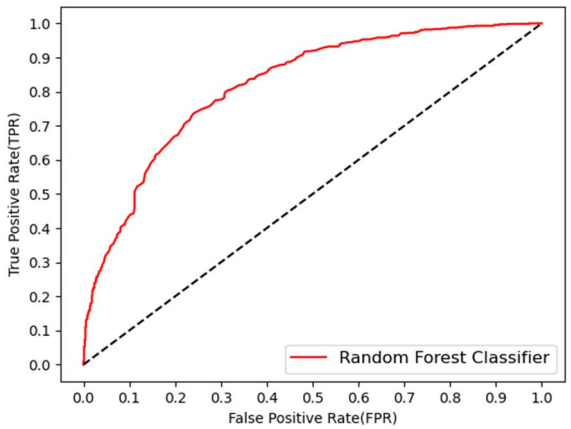

Fig. 5. ROC curve of Random Forest Classifier

The past works on accident detection by [19], [27] and [20] were chosen for comparison since they have used a dataset similar to that used in the presented framework in terms of the diversity of conditions in which the accidents were captured.

TABLE III. COMPARISON WITH PAST WORKS

Approach	DR %
Vision based model (ARRS) [27]	50
Computer Vision-based Accident Detection in Traffic Surveillance [19]	71
Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder [20]	77.5
Presented framework	**78.2**

The proposed framework achieves a detection rate of 78.2% which surpasses [19], [27] and [20]. Even though this work does not exceed the performance of [17], it does not lead to overfitting as in case of the latter. The obtained performance reported is due to the use of DETR which has helped to produce highly accurate detections which in turn has improved the performance of the downstream classification task.

V. CONCLUSION AND FUTURE SCOPE

The presented work serves as an accident detection framework for detecting road traffic accidents in various weather conditions such as day, night, foggy, snowy or dusty conditions. The use of the new state-of-the-art Detection Transformer has empirically proved to be a remarkable improvement in accident detection using CCTV videos. The Random Forest classifier was used downstream to aid the classification task. A detection rate of 78.2% was achieved with low latency compared to past works. A drawback of the DETR as of now is that it finds difficulty in localizing diminished objects.

Future works which can be carried out to increase the detection rate are as follows:

- Image enhancement techniques like Retinex or CLAHE (Contrast Limited Adaptive Histogram Equalization) can help object detection algorithms handle frames captured in low-visibility conditions with better accuracy
- Stacking ensembles of different Random Forest classifiers which makes an overall prediction based on a weighted sum of different predictions from the classifiers can increase the accuracy of the final result

REFERENCES

[1] WHO, "Road traffic injuries," Feb 7 2020. Accessed on: July 10, 2020. [Online], Available: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries

[2] Statista Research Department, "Number of Deaths due to road accidents across India from 2005 to 2018," Sept 2019. Accessed on: July 10, 2020. [Online]. Available: https://www.statista.com/statistics/746887/india-number-of-fatalities-in-road-accidents/

[3] Rahul Chhabra, Financial Express, "40% Highway accidents occur due to drivers dozing off," July 30, 2019. Accessed on: July 10, 2020. [Online]. Available: https://www.financialexpress.com/india-news/40-of-highway-accidents-occur-due-to-drivers-dozing-off/1659901/

[4] A. Chayeb, N. Ouadah, Z. Tobal, M. Lakrouf and O. Azouaoui, "HOG based multi-object detection for urban navigation," *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, *Qingdao*, pp. 2962-2967, 2014.

[5] Bochkovskiy, Alexey et al., "YOLOv4: Optimal Speed and Accuracy of Object Detection." *ArXiv* abs/2004.10934 (2020): n. Pag.

[6] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA*, vol.1, pp. 886-893, 2005.

[7] L. Jiao *et al*., "A Survey of Deep Learning-Based Object Detection," *IEEE Access*, 7, pp. 128837-128868, 2019.

[8] R. Girshick, "Fast R-CNN," *2015 IEEE International Conference on Computer Vision (ICCV), Santiago*, pp. 1440-1448, 2015.

[9] S. Ren, K. He, R. Girshick and J. Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 39(6):1137-1149, 2017.

[10] J. Redmon, S. Divvala, R. Girshick and A. Farhadi, "You Only Look Once: Unified, Real-Time Object Detection," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Las Vegas, NV,* pp. 779-788, 2016.

[11] W. Liu *et al*., "SSD: Single Shot MultiBox Detector," *European Conference on Computer Vision (ECCV)*, Amsterdam, The Netherlands, pp. 21-37, 2016.

[12] Redmon, Joseph & Farhadi, Ali, "YOLOv3: An Incremental Improvement," 2018.

[13] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov and S. Zagoruyko, "End-to-End Object Detection with Transformers,"2020. [Online]. Available: arXiv:2005.12872.

[14] S. Kumaran, D. Dogra and P. Roy, "Anomaly Detection in Road Traffic Using Visual Surveillance: A Survey,"2019. [Online]. Available: arXiv:1901.08292.

[15] N. Patil and P. Biswas, "Global abnormal events detection in surveillance video — A hierarchical approach," *2016 Sixth International Symposium on Embedded Computing and System Design (ISED)*, Patna, India, pp. 217-222, 2016.

[16] Kaltsa, Vagia & Briassouli, Alexia & Kompatsiaris, Ioannis & Hadjileontiadis, Leontios & Strintzis, Michael, "Swarm Intelligence for Detecting Interesting Events in Crowded Environments," *IEEE transactions on image processing*, 24(7):2153-2166, 2015

[17] Wang, Chen & Yulu, Dai & Zhou, Wei & Geng, Yifei, "A Vision-Based Video Crash Detection Framework for Mixed Traffic Flow Environment Considering Low-Visibility Condition," *Journal of Advanced Transportation*, 2020.

[18] K. He, G. Gkioxari, P. Dollár and R. Girshick, "Mask R-CNN," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2), pp:386-397, 2020.

[19] E. P. Ijjina, D. Chand, S. Gupta and K. Goutham, "Computer Vision-based Accident Detection in Traffic Surveillance," *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT),* Kanpur, India, pp.1-6, 2019.

[20] D. Singh and C. K. Mohan, "Deep Spatio-Temporal Representation for Detection of Road Accidents Using Stacked Autoencoder," *IEEE Transactions on Intelligent Transportation Systems*, 20(3): 879-887.

[21] L. Breiman, "Random Forests," *Machine Learning*, 45(1), pp:5-32, 2001.

[22] K. He, X. Zhang, S. Renand J. Sun, "Deep Residual Learning for Image Recognition," *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Las Vegas, NV, USA, pp.770-778, 2016.

[23] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, L. Kaiser and I. Polosukhin "Attention Is All You Need," 2019. [Online]. Available: arXiv: 1706.03762.

[24] I. Bello, B. Zoph, Q. Le, A. Vaswani and J. Shlens, "Attention Augmented Convolutional Networks," *IEEE International Conference on Computer Vision, Seoul (ICCV),* South Korea, pp. 3285-3294, 2018.

[25] J. Ba, J. Kiros and G. Hinton, "Layer Normalization," 2016. [Online]. Available: arXiv:1607.06450.

[26] A. Shah, J. Lamare, T. Nguyen-Anh and A. Hauptmann, "CADP: A Novel Dataset for CCTV Traffic Camera based Accident Analysis," *IEEE International Conference on Advanced Video and Signal based Surveillance (AVSS),* Auckland, New Zealand, pp.1-9, 2018.

[27] Y. Ki and D. Lee, "A Traffic Accident Recording and Reporting Model at Intersections," *IEEE Transactions on Intelligent Transportation Systems*, 2(8): 188 – 194, 2006.

[28] R. Girshick, J. Donahue, T. Darrell and J. Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation," *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Columbus, OH, pp. 580-587, 2014

[29] Loshchilov, Ilya and Frank Hutter, "Decoupled Weight Decay Regularization", *ICLR*, 2019.

[30] J. Deng, W. Dong, R. Socher, L. Li, Kai Li and Li Fei-Fei, "ImageNet: A large-scale hierarchical image database," *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR),* Miami, FL, USA, pp.248-255, 2009

[31] A. Sherstinsky, "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network," 2018. [Online]. Available: arXiv:1808.03314.