PAPER ID: ICAIS 513
TITLE: AUTOMATIC ROAD ACCIDENT DETECTION USING DEEP LEARNING

Authors:
Gopika Rani N
Adarsh G
Ajay Deepak P M
Elanthamil R
Jeevan Krishna K V
Nirmal M

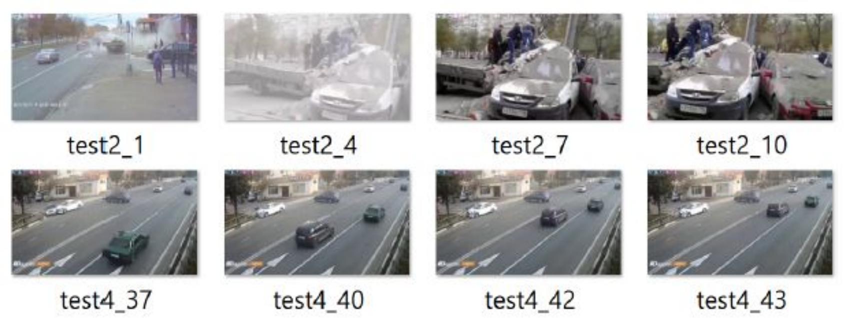
Abstract

- Accident detection is a critical task in improving road safety and deep learning techniques have been extensively employed to detect accidents from data sources.
- The first step is the collection of a diverse dataset of accident and non-accident samples which is used to train deep learning models.
- Deep learning is emphasized as a powerful tool for accident detection, utilizing diverse datasets to train models capable of recognizing subtle accident-related features. These models trigger critical responses like alerts and emergency service notifications upon identifying accidents.
- The integration of the Simple Mail Transfer Protocol (SMTP) for email alerts is a crucial feature, ensuring rapid communication with relevant stakeholders and emergency services when accidents occur.

Dataset

- The dataset contains various photos that were taken from actual events.
- Training, testing and validation are organized into three separate subsets, each containing a mix of accident and non-accident scenarios.
- The dataset contains a large number of pictures showing the crucial moments before, during, and after accidents, along with pictures showing non-accidental situations.
- The dataset contains 990 files in .jpeg format.
- The Training folder contains 422 non-accident files and 369 accident files.
- The Test folder contains 47 accident images and 54 non accident images to test out the model that has been created.
- The Validation folder contains 46 accident images and 52 non accident images.

1. **Unlabeled Data:**
   Unlabeled data for accident detection refers to raw, not annotated information, lacking specific indications of accidents. The raw data in Figure is unlabeled. This data has to be converted into bounding box vectors to train the YOLO model.

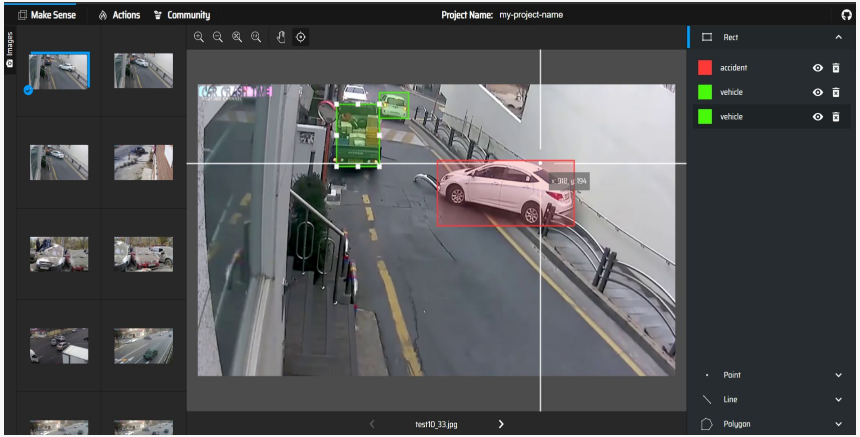


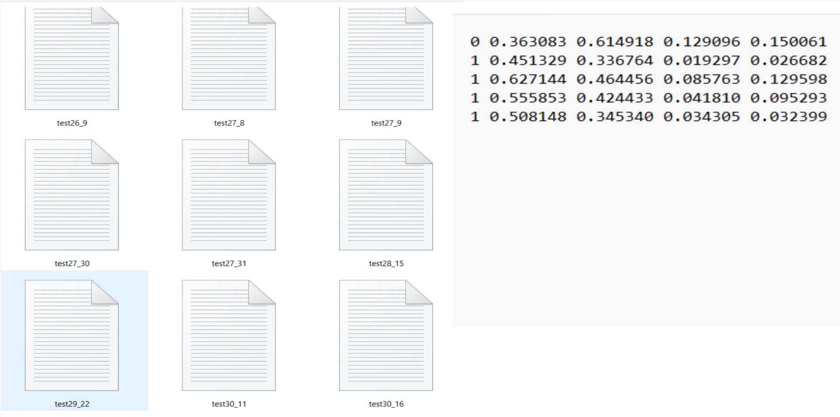
2. Drawing Bounding Boxes:

Drawing bounding boxes for accident detection involves visually describing the region of interest, typically an accident or no accident, within a CCTV video frame. This method is widely employed to annotate and train deep learning models for precise accident detection. In this context, the model is tasked with detecting bounding boxes around no accident and accident scenes.

3. Labeled Data:

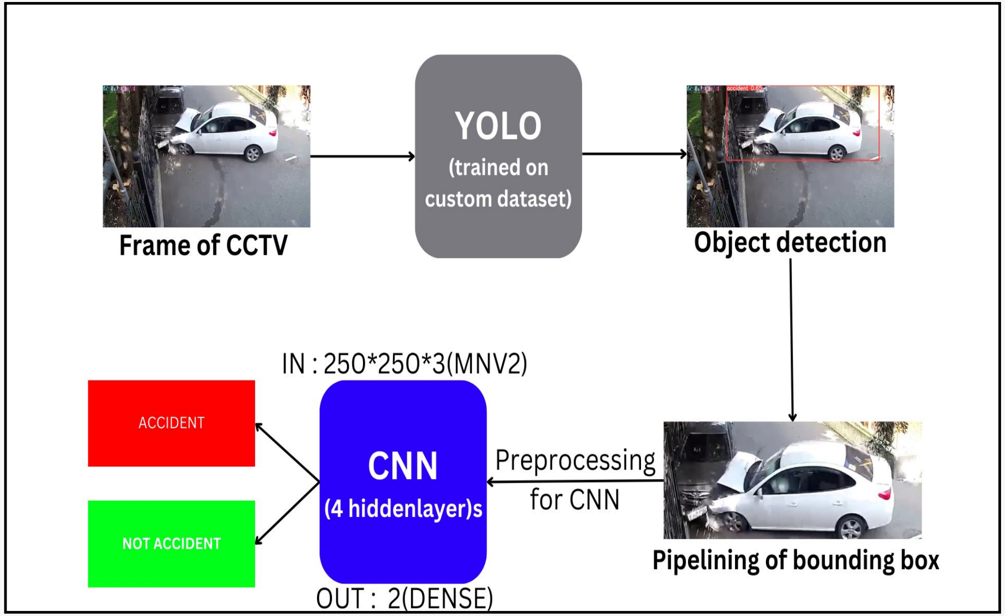
Each image in the dataset is associated with a corresponding text file containing vital information. Each label indicates whether the object represents an accident (denoted as "0") or no accident (denoted as "1"). Following this classification, the next two values specify the coordinates of the bounding boxes top-left corner, denoting the x and y coordinates. Subsequently, the width and height of the bounding box are provided as the next two values, encapsulating the object's spatial dimensions within the image.





System Architecture

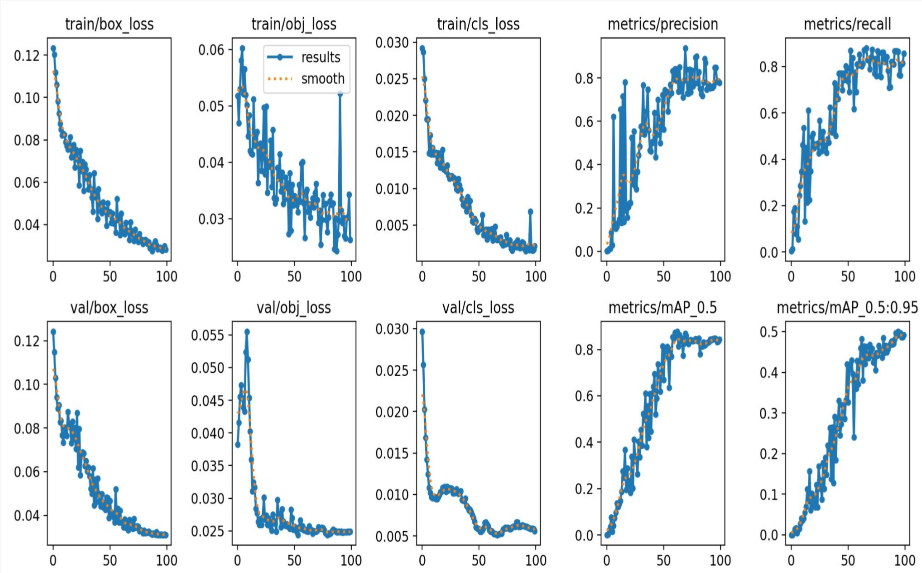
- The initial input to the YOLO object detection algorithm is the CCTV video stream, in which each frame is processed sequentially. This YOLO model is trained on a custom dataset. This training helps YOLO model to detect objects in each frame from the CCTV video stream.
- After successful identification of objects in the frame by YOLO model, the objects are detected for accidents. If the YOLO model detects no objects as an accident the frame is not processed further and classified as "No Accident". If the YOLO model detects objects as an accident, the frame is processed further with pipelining to the CNN model.
- The bounding box of the objects detected as an accident from the frame is extracted and passed to the CNN model. These accident detected objects are then fed into a 6 - layer CNN model. This CNN model has been trained on a diverse dataset, encompassing accident and no accident scenarios. This training enables CNN model to make decisions when classifying the bounding boxes of the accident detected objects as "Accident" or "No Accident".



Architecture for accident detection

YOLO Model

The YOLO model provides exceptional speed and efficient object localization of CCTV video frames. Unlike traditional methods that require multiple passes through an image, the YOLO model uses a single-pass approach that significantly reduces processing time for accident detection.



Results of the YOLO model during the training phase

Convolutional Neural Network Model

I. **Input Layer**

It is the layer to which we give the inputs. The number of neurons in this layer is the number of pixels in the image. In this case, the input is 250*250*3 which is 187500. Here, the 250 and 250 specify the height and width, 3 is the depth, batch size is None. The output is (8, 8, 1280) which is height, width and depth respectively.

II. **Convolutional Layer 1**

The output of this Input Layer is given to Convolutional Layer 1. The kernel is convolved across the height and width of the input volume. The output which is an activation map is passed to the Convolutional Layer 2. The input to this layer has a shape of (8, 8, 1280) and the output has a shape of (6, 6, 32).

III. **Convolutional Layer 2**

The output of this Convolutional Layer 1 is given to Convolutional Layer 2. The input is of shape (6, 6, 32). After further convolution the output is of shape (4, 4, 64).

IV. **Convolutional Layer 3**

The output of this Convolutional Layer 2 is given to Convolutional Layer 3. It performs further convolution with the help of the kernel, the input received by this layer is of shape (4, 4, 64) and the output obtained is of shape (2, 2, 128).

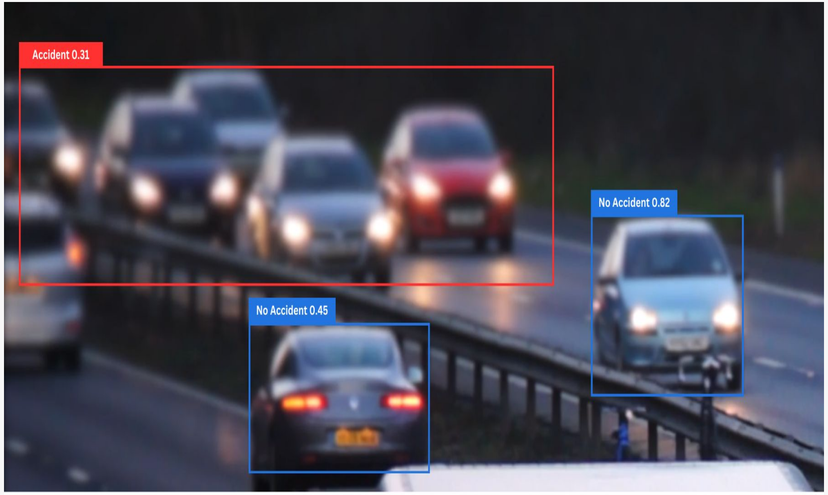
V. **Flatten Layer**

The flatten layer is used to convert the feature map that it received to a format that the dense layer can understand. It converts the multi dimensional array of the feature map to a one dimensional array. In this case the input is of shape (2, 2, 128). It converts this input to output of shape 512 as a one dimensional array.

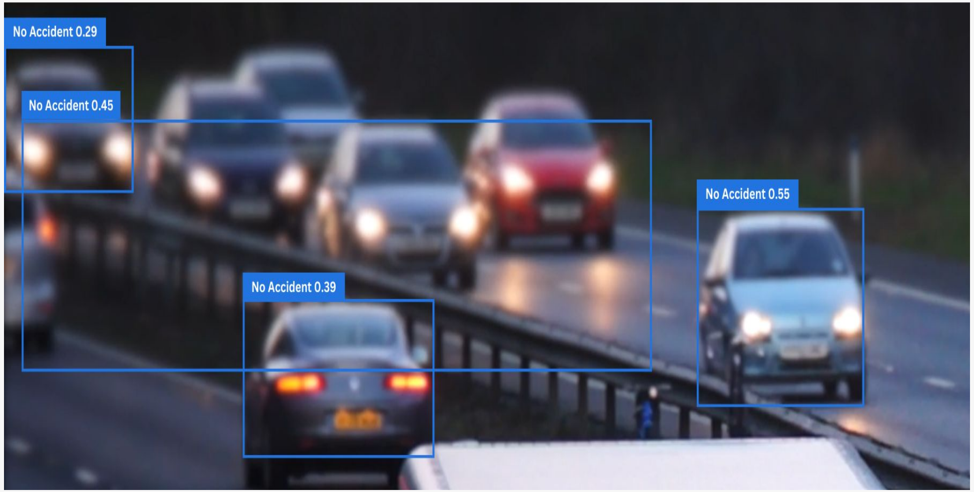
VI. **Dense Layer**

The dense layer is used for classification based on the inputs received from the previous layer. In this case, it classifies whether it is an “Accident” or “No Accident”. It gets an input of size 512 as a one dimensional array and classifies it into two classes(Accident or No Accident). Therefore, output is classified into 2 classes.

Results



False positive detected by the YOLO model



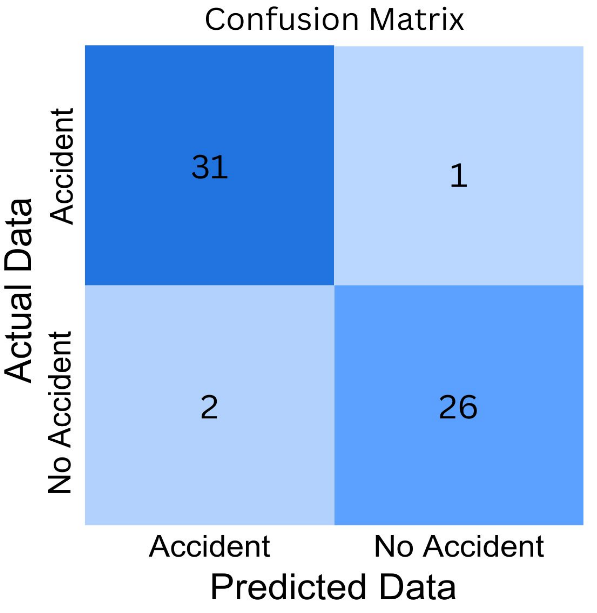
Correction of Classification by the CNN model

Sequence of Frames	Class	Time taken for detection
1	No Accident	224.4 ms
2	Accident	923 ms
3	No Accident	233.6 ms
4	Accident	924 ms

Time taken for each frame by the Proposed model for no accident and accident detection

	Precision	Recall	F1 Score	Support
Accident	0.94	0.97	0.95	32
No Accident	0.96	0.93	0.95	28
accuracy			0.95	60
macro avg	0.93939	0.96875	0.95384	60
weighted avg	0.95	0.95	0.95	60

Classification report of the proposed model



The model is tested on 60 images. The confusion matrix of the results predicted by the proposed model is shown in the above Figure. Analysis of Figure reveals that the model predicted 31 “Accident” cases and 2 “No Accident” cases as “Accident” cases. Furthermore, there were 1 false negative case and 26 true negative cases.

Figure shows the notification received when an Accident is detected with the help of the SMTP. The message includes location, co-ordinates, date and time of the accident.

Comparison with other Deep Learning Algorithms

The comparison between the proposed model and other deep learning models that includes FI_C0166000_EfficientNetB1, CNN (9 layers), and CNN with LSTM is performed and shown in the table. Among other deep learning models the CNN with LSTM model provided a maximum accuracy of 91.38%, while the proposed model achieved a better accuracy of 95%. It can be inferred that the proposed model achieves the highest accuracy when compared to other deep learning models. Overall, the proposed model shows a significant improvement in accuracy when compared to other deep learning models. Thus, from the table, it can be inferred that the proposed model provides a better performance and better prediction compared to the other deep learning models.

Models	**Accuracy**
FI_C0166000_EfficientNetB1	0.88
CNN (9 layers)	0.85
CNN with LSTM	0.9138
Proposed model	0.95

References

1. Tiago Tamagusko, Matheus Gomes Correia, Minh Anh Huynh, Adelino Ferreira, (2022) Deep Learning applied to Road Accident Detection with Transfer Learning and Synthetic Images, Transportation Research Procedia, vol. 64, pp. 90-97, doi: 10.1016/j.trpro.2022.09.012
2. G. Rajesh, A. R. Benny, A. Harikrishnan, J. Jacob Abraham, N. P. John, (2020) A Deep Learning based Accident Detection System, International Conference on Communication and Signal Processing (ICCSP), Chennai, India, pp. 1322-1325, doi: 10.1109/ICCSP48568.2020.9182224
3. D. K. Yadav, Renu, Ankita, I. Anjum, (2020) Accident Detection Using Deep Learning, 2nd International Conference on Advances in Computing, Communication Control and Networking (ICACCCN), Greater Noida, India, pp. 232-235, doi: 10.1109/ICACCCN51052.2020.9362808
4. H. Shi, C. Liu, (2018) A New Foreground Segmentation Method for Video Analysis in Different Color Spaces, 24th International Conference on Pattern Recognition (ICPR), Beijing, China, pp. 2899-2904, doi:10.1109/ICPR.2018.8545500
5. G. Liu et al, (2022) Smart Traffic Monitoring System Using Computer Vision and Edge Computing, IEEE Transactions on Intelligent Transportation Systems, vol. 23, no. 8, pp. 12027-12038, doi:10.1109/TITS.2021.3109481
6. H. Ghahremannezhad, H. Shi, C. Liu, (2020) Automatic Road Detection in Traffic Videos, IEEE Intl Conf on Parallel & Distributed Processing with Applications, Big Data & Cloud Computing, Sustainable Computing & Communications, Social Computing & Networking (ISPA/BDCloud/SocialCom/SustainCom), Exeter, United Kingdom, pp. 777-784, doi:10.1109/ISPA-BDCloud-SocialCom-SustainCom51426.2020.00123
7. H. Ghahremannezhad, H. Shi, C. Liu, (2020) A New Adaptive Bidirectional Region-of-Interest Detection Method for Intelligent Traffic Video Analysis, IEEE Third International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), Laguna Hills, CA, USA, pp. 17-24, doi: 10.1109/AIKE48582.2020.00012
8. H. Ghahremannezhad, H. Shi, C. Liu, (2020) Robust Road Region Extraction in Video Under Various Illumination and Weather Conditions, IEEE 4th International Conference on Image Processing, Applications and Systems (IPAS), Genova, Italy, pp. 186-191, doi:10.1109/IPAS50080.2020.9334959
9. H. Shi, H. Ghahremannezhadand, C. Liu, (2020) A Statistical Modeling Method for Road Recognition in Traffic Video Analytics, 11th IEEE International Conference on Cognitive Infocommunications (CogInfoCom), Mariehamn, Finland, pp. 000097-000102, doi:10.1109/CogInfoCom50765.2020.9237905
10. Charan Kumar C, (2020) Accident Detection From CCTV Footage, Kaggle, https://www.kaggle.com/datasets/ckay16/accident-detection-from-cctv-footage