Accidents Prediction on Pedestrian Using CNN

Lakshmi Keerthi Ayyagari

Northern Illinois University

Professor Dr. Mona Rahimi

Abstract

This research provides an accident prediction method based on the cause of pedestrian accidents. The paper described the neural network and its parameters. Accidents involving pedestrians are becoming more prevalent, and under the guise of accidents, individuals commit crimes. Accident data is growing year after year, and the majority of the time, accidents result in death. The goal of this article is to determine the cause of pedestrian accidents, grasp the CNN method, and learn how to handle and work with video data as well as Image Annotation to label the video/Image data. At each epoch, the CNN model delivered a high degree of accuracy.

Introduction

The Paper introduces a deep learning CNN model for CCTV video accident detection on pedestrian. Recently, CCTV project on India's street has been launched to improve citizen's safety. This CCTV installed for crime prevention and traffic information collection. The traffic or pedestrian information collection increasing every year [1]. All CCTVs are monitored by the government people mostly and they can take action when any suspicious situation happens. Various suspicious situations exist, but when a traffic or pedestrian accidents happens then destruction is usually big and immediate action is needed. Pedestrian accidents may occur due to human's mistake or due to environmental change. If the government people can monitor CCTV and find out the reason of pedestrian accidents, then efficiency of CCTV control system could be increased [4]. To detect the pedestrian detection and its reason, system applying different video and image processing techniques and deep learning techniques to increase the efficiency of CCTV through automatic detection of the occurrence of pedestrian accidents and can be stop the second accident by urgent response or making urgent rule for pedestrian during bad environment. In this paper, CCTV video footages converted to image and all the processing has been done on images, and accident prediction with reason is proposed by the CNN deep learning neural network model. A deep learning technique is growing fast from past few decades. Deep learning model is a powerful tool because of its ability to handle the big data and implement complex mathematical algorithm on the big data[5][7]. Accident prediction and its reason prediction will help the government to understand the most reason of having accidents on road, number of accidents happened in the road daily. Due to this prediction and results of accidents reasons, government can create new rule for traffic/ car drivers/roads or can update the previous rule with some changes. Government can provide more safety or more instructions to the drivers during bad weather or during rainy seasons.

Research Problem

The objective of this paper is to classify the situations into three categories 'Accidents due to human error', 'Accidents due to environment error', and 'non-accident' situations, classification and prediction of three categories are the main aim of the paper. Aim is to understand the process of CNN algorithm and its parameters.

Dataset

The video of CCTV footage is taken from the YouTube accident compilation Video. Pedestrian video dataset is selected from the Yotube.com (Indian Road accidents compilation 2021 - YouTube).

Methods

To solve the research problem Image annotation and CNN method has been implemented. In order to implement the CNN method some libraries are used like 'Numpy', 'pandas', 'Matplotlib', 'Pixellib', 'tensorflow', 'keras', 'Open-cv', etc.

Python Libraries

In this paper some high and state of the art libraries has been used in python like openCV, Tensorflow, and keras.

Open-CV

Open-cv is the big open-source library in python for computer vision, Machine learning, and image processing, real time operation, etc. By using openCV, Video processing, image processing can be possible to identify the objects, faces,etc. OpenCV is used for object detection, medical image analysis, Face recognition, etc. In python one can install it using '!pip install opencv-python' or 'conda install –c conda-forge opencv-python'.In this paper opencv 4.5.4 version has been used for video and image preprocessing[16]. Using Opencv Libraray Converted CCTV Video of accidents on Indian pedestrian data to images and stored the images to a folder.

Tensorflow

Tensorflow is a biggest library just like openCV, Tensroflow is used for hard mathematical computation, in backend numpy library is the supportive library for tensorflow, Tensorflow has some version, in this paper Tensorflow 2.7.0 version has been used. Keras is the sublibrary of tensorflow. Keras is the high level API of Tensorflow. It is a deep learning framework in python. Tensorflow has the backend algorithm like neural network, deep learning algorithms, etc. Keras makes works easier as tensorflow is working in backend. Tensorflow can complete the 50000-6000 steps in building model just in 15-20 minutes of processing the model, however keras is able to complete the 50,000-60,000 steps in building model in more than 2 hours of processing the model. In keras performance is quite slow but in tensorflow it does the high performance on model. Tensorflow and Keras library is used in Image data generation and CNN model[11][12].

Image Annotation

For identifying converted images of video, in python labelImg library exist but one cannot use this library directly. Need to use some command in command prompt an dhave to create environment for labelimage. Steps to set the environment for labelimg is:

Step-1] Download Zip folder from the Github tzutalin/labelImg: 🖉 LabelImg is a graphical image annotation tool and label object bounding boxes in images (github.com) . Setup the LabelImg folder and locate it into C drive user location.

Step-2] open Anaconda prompt, Activate Virtual environment. Install 'PyQt' uding 'conda install pyqt=5'

Step-3] Install lxml using 'conda install –c anaconda lxml.

Step-4] type cd in command prompt to go inside the labelimg folder then run labelImg-master to go inside the folder.

Step-5] Run command ‘pyrcc5 – resources.py resources.qrc’ to create resources.py file and resources.qrc file and move this two file into libs folder in the system.

Step-6] Run command ‘Python labelimg.py’ to open the labelimg environment or tool for image annotation.

By using this library one can perform the steps of image annotation, but used another way to label or annotate the images, downloaded the ‘Image Annotation Lab’ Software and classified the images into three Different categories ‘Accident due to human error’, ‘Accident due to environment error’, and ‘Non-Accident’. During Image annotation, images are masked into some specific colors, for ‘Accident due to human error’ images color is ‘Blue’, ‘Accident due to environment color is ‘Yellow’, and ‘Non-accident’ image color is ‘Green’.

Fig1.0 : Labelled Image of Accident due to Environmental Error

Classified and saved all the imges manually into 3 different folder named as ‘AccidenthumanError’, ‘AccidentEnvironment Error’, and ‘NonAccident’ based on the colors (Refer Fig 1.0,1.1, and 1.2). Fig 1.0 shows that accidents happened due to environment error, dut to rain roads becomes sleepary for the vehicles and one must have to drive carefully during rainy season.

Fig 1.1: Labelled Image of Accident due to Human Error

Accident has done on road due to mistakes of two car drivers, Accident happened in image is due to human mistakes (Refer Fig 1.2).

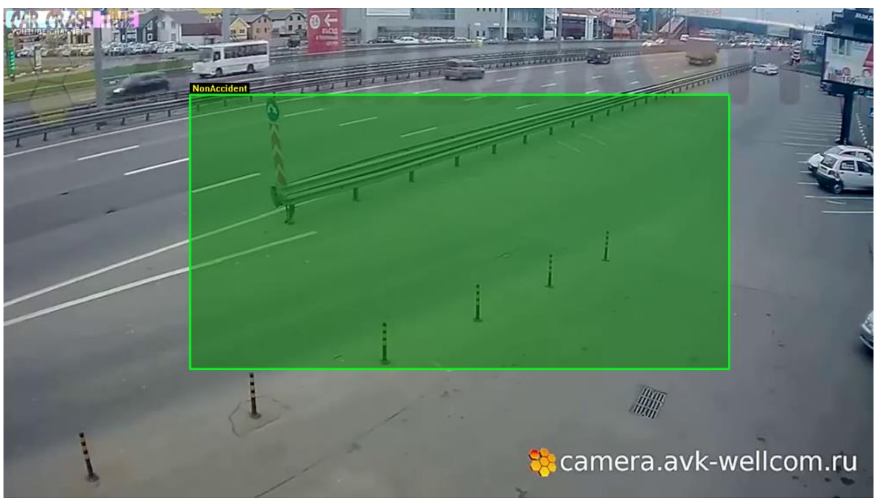

Fig 1.2: Labelled Image of Non-Accident

No accident has done road is empty and everything seems smooth going, but environment in the images seems not good (Refer Fig 1.2).

Using ‘Splitfolders’ library in python splitted the ‘Accidenthumanerror’, ‘AccidentEnvironmenterror’, ‘Non-accident’ folder to ‘train’, ‘test’ and ‘val’ folder. In every folder of all the three folder is splitted in 70% into train folder, 20% into test and 10% into val folder.

Image data Generator

To use Image data Generator keras has been used where tensorflow was working in backend. Image augmentation is a great way to convert image data to numerical data in order to implement in the algorithm. Keras image data generator has a host of different augmentation techniques like standardization, rotation, shifts, flips, etc. Keras image data generator designed to provide real time data augmentation is the real application of keras image data generator. Image data generator needs very low memory usage and this is the application of image data generator. It stores the images to numpy arrays and can proceed further for implementation of model [11][12][15].

CNN (Convolutional Neural Network)

CNN is a convolution neural network, a deep learning neural network algorithm. CNN algorithm is the first developed and used in the year around 1980's, at that time CNN was only able to recognize the hand written digits. Nowadays CNN mostly used for Image data analysis. CNN uses a special technique in place of matrix multiplication known as convolution. Convolution is a mathematical operation that applied on two functions which creates a third function that shows the process of shape of one is modified by the other shape. Every image is built using only three colors i.e RGB (Red, Green, and Blue). An RGB image is a matrix of pixel values that has three planes, however grayscale image is the same but has the single plane[3][5][21]. CNN are made up of multiple layers of artificial neurons. It is a mathematical function that calculates the weighted sum of multiple inputs and outputs of an activation value. Each layer in CNN generates the various activation functions which passed to the next layer. First layer mostly extracts the basic features like diagonal or horizontal edges. First layer output passed to the next layer that detects the more complex features like combinational edges [21]. Moving deeper to the network will be able to identify more and more complex features like faces, objects, etc.

Pooling layers work is to reduce the spatial size of convolved feature just like the convolutional layer. Two types of pooling layer is exist, one is 'Average Pooling' and 'Max Pooling'. In max pooling, one must have to find the maximum value of a pixel from a part of the image, this works as noise suppressant. It removes the noisy activation all at once and works as de-noising with dimension reduction. Average pooling provides the mean value of all the values from the part of the images that is covered by the kernel. Average pooling works as a dimensionality reduction as a noise suppressor. This interprets that Max pooling works much better than Average pooling [9].

Batch Size

During training a neural network model updates the metric of the model using some calculations in the data. If the size of the data is large this might need a lot of time and consume a lot resources to complete the training. To save the time and resources, batch of data is created also known as batch data processing[3]. It is important to use batch size if data size is very big and not able to fit the data into machine's memory. Batch size is the number of items from the training data. In this paper selected batch size is 100 that means average error and weights update every 100 items. Total train size of the dataset is 1370 and batch size setup is 100, The algorithm takes first 100 samples from the train dataset and train the network, next algorithm takes another 100 samples from the train dataset and trains the network again, and so on. These process will continue until all the samples not get propagated through all the network.

Epochs

Whenever batch of data pass through the neural network, model or train data completed in one iteration. Epoch calculated as, epoch= batch size* number of iterations[5].

Activation Function

During the model training 'ReLu', and 'Softmax' activation function has been used. Actiavtion function is used to determine the output of the neural network[5][6][12].

- **ReLu**

  ReLu is a Rectified Linear Unit Activation function, and it is the mostly used activation function. This functions ranges are from 0 to infinity, in this function all the negative values converted to zero and rate of conversion is very fast.

  Mathematical formula for ReLu function, $f(x)= max(x,0)$.

  This function converts negative value to zero, or if value is positive then value will be same (No change in values).

- **Softmax**

  Softmax activation function calculates the relative probabilities, it uses the value to determine the final probability value. Softmax function returns the probability of each class. Equation for the softmax activation function is:

$$\text{Softmax}(Z_i)=\frac{e^{Z_i}}{\sum e^{Z_j}}$$

  Here, Z represents the value from the neurons of the output layer. The exponential acts as a non-linear function and these values are splitted by the sum of exponential values to normalize and then convert it into probabilities[3][5].

Optimizer

In this paper, ADAM optimizer is used which is the most used optimizer.Optimizers are used to change the attributes of a neural network to reduce the losses [5]. ADAM is also known as Adaptive Moment Estimation. From the estimates of first and second moments of gradients, adaptive learning rate for every parameter is calculated. ADAM is a combination of Adagrade that process good on sparse gradients and RMSprop. Instead of simple average, adam works with exponential moving average of the gradients to scale the learning rate. ADAM optimizer requires very less memory and it is computationally efficient [12]. Adam algorithm first updates the exponential moving averages of the gradient(mt) and the squared gradient(vt) which is the estimates of the first and second moment. Hyper-parameters $\beta_1, \beta_2 \in [0,1)$ control the decay rates of the moving averages.

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) {g_t}^2$$

Final Mathematical function for ADAM optimizer is,

$$\theta_{t+1} = \theta_t - \frac{\gamma \widehat{m}_t}{\sqrt{\widehat{v}_t + \varepsilon}}$$

Loss

Sparse categorical cross entropy has been used in CNN model. When values are integers, sprse_categorical_crossentropy is used, advantage of sparse categorical cross entropy is it saves the time as well as computation in a memory because this loss function uses a single integer for a class instead of a whole vector.

Accuracy

Accuracy is a metric that interprets about the fitness of the model. Accuracy speaks about the accuracy of model, that is how accurate model can provide the correct predicted results [1].

Limitations

CNN method has some limitations, this method just identifies the objects not more than that, For accidents reasons detection CNN method is not suitable, this can detect the only objects like whether object is car, person, cat, etc. And method requires labels. This method cannot describe about the situation in the image. Accident happened or not CNN method classified that but the reason of Accident cannot be identified by the CNN method without the label or pretrained labeled images[18].

Results

In CNN model, maximum pooling with activation function ‘relu’ and ‘softmax’ is used.
Optimizer is ADAM with ‘sparse_categorical_crossentrpy’ loss function is used.

```
In [14]: from tensorflow.keras import layers
         Cnn = tf.keras.models.Sequential([
           layers.BatchNormalization(),
           layers.Conv2D(32, 3, activation='relu'),
           layers.MaxPooling2D(),
           layers.Conv2D(64, 3, activation='relu'),
           layers.MaxPooling2D(),
           layers.Conv2D(128, 3, activation='relu'),
           layers.MaxPooling2D(),
           layers.Flatten(),
           layers.Dense(256, activation='relu'),
           layers.Dense(len(class_names),activation= 'Softmax')
         ])

         Cnn.compile(optimizer='adam',loss='sparse_categorical_crossentropy', metrics=['accuracy'])

In [15]: retVal = Cnn.fit(training_ds, validation_data= validation, epochs = 10)

         Epoch 1/10
         7/7 [==============================] - 454s 64s/step - loss: 8.3328 - accuracy: 0.4421 - val_loss: 1.2891 - val_accuracy: 0.529
         5
         Epoch 2/10
         7/7 [==============================] - 403s 57s/step - loss: 0.4993 - accuracy: 0.8189 - val_loss: 0.5369 - val_accuracy: 0.779
         0
         Epoch 3/10
         7/7 [==============================] - 206s 29s/step - loss: 0.1415 - accuracy: 0.9608 - val_loss: 1.1284 - val_accuracy: 0.691
         5
         Epoch 4/10
         7/7 [==============================] - 327s 49s/step - loss: 0.1344 - accuracy: 0.9543 - val_loss: 0.2297 - val_accuracy: 0.923
         4
         Epoch 5/10
         7/7 [==============================] - 290s 40s/step - loss: 0.0694 - accuracy: 0.9853 - val_loss: 0.5899 - val_accuracy: 0.698
         0
         Epoch 6/10
         7/7 [==============================] - 356s 54s/step - loss: 0.0367 - accuracy: 0.9918 - val_loss: 0.3749 - val_accuracy: 0.857
         8
         Epoch 7/10
         7/7 [==============================] - 349s 48s/step - loss: 0.0167 - accuracy: 0.9967 - val_loss: 0.3586 - val_accuracy: 0.884
         0
         Epoch 8/10
         7/7 [==============================] - 371s 55s/step - loss: 0.0075 - accuracy: 1.0000 - val_loss: 0.3087 - val_accuracy: 0.901
         5
         Epoch 9/10
         7/7 [==============================] - 374s 53s/step - loss: 0.0035 - accuracy: 0.9984 - val_loss: 0.5706 - val_accuracy: 0.855
         6
         Epoch 10/10
         7/7 [==============================] - 264s 38s/step - loss: 0.0027 - accuracy: 1.0000 - val_loss: 0.2337 - val_accuracy: 0.934
         4
```

Fig 1.3: CNN Model

As per fig 1.3, CNN model gives the 93.40% accuracy with 23.37% loss on validation data and consequently 100% accuracy on trained data with 0.27% loss in 10th epoch. As per the Fig 1.3 results of accuracy on trained and validation data is very good and it can be interpretable that at 10th epoch validation data correctly predict the data 93.4% times.

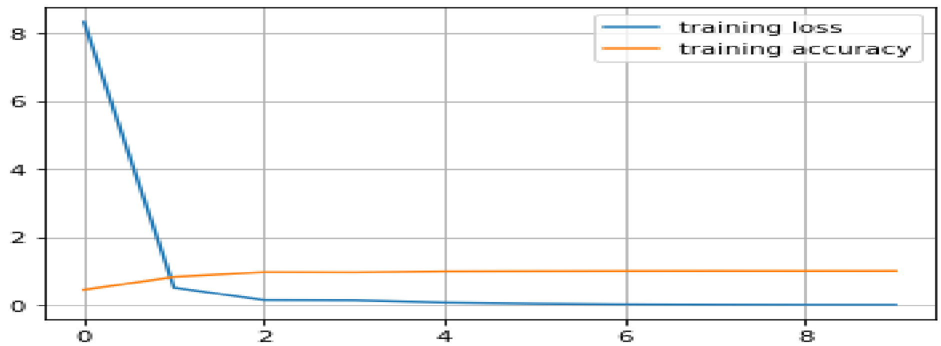

Fig 1.4: Training loss And Training Accuracy Plot

Fig 1.4, explains the opposite relation between loss and accuracy, when loss of parameter is high accuracy of model was very low, and suddenly loss of parameter gets low and accuracy of train data is got increased. As per fig 1.4, loss and accuracy is parallel to each other after some point i.e 2.

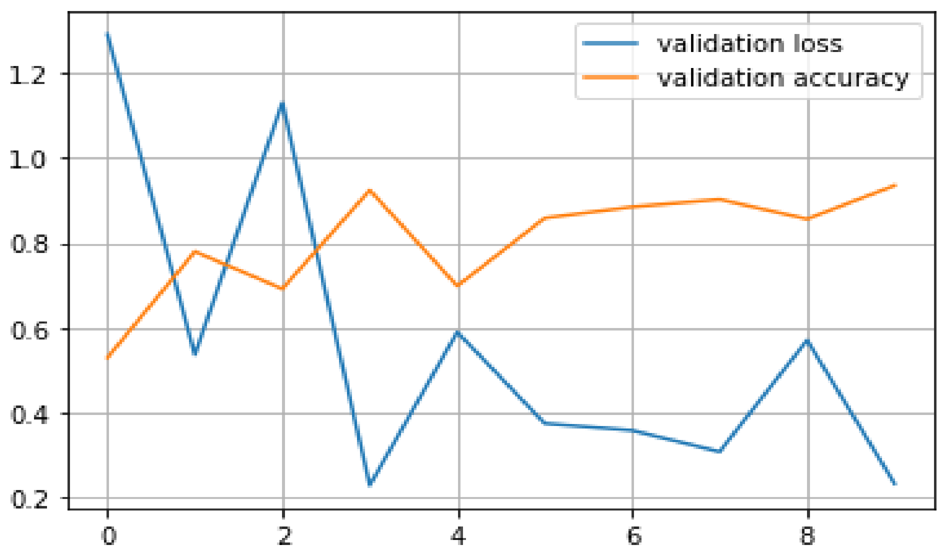

Fig 1.5: Validation Loss And Validation Accuracy Plot

Fig 1.5, explains about the validation loss and validation accuracy. In start, Validation loss is high and validation accuracy is low, but after some point validation accuracy is increased and validation loss got started decreasing.

These increment and decrement of loss and accuracy explained about the machine or model learning. In 10 epoches, model reached good accuracy level with very less parameter loss.

Fig 1.7: Predicted Images Accidents

Fig 1.7 is the predicted images, these images are predicted by the trained CNN model. In Fig 1.7, Some images are incorrectly predicted but most of the images are correctly predicted by the model.

Discussion & Conclusion

Video Analysis is a very critical task to perform using any programming language but Deep learning methods makes it somehow easier. There are many object detection method in deep learning like , R-CNN, yolo, etc.[2][3][6]. Identifying the situations from the muted video is not possible because one cannot process the audio and can understand from the audio that at which situation or moment what incidence happened. In this paper we had taken muted youtube video, converted that into the images; now processing the situation from the images is not easy from any of the above stated algorithm. So, Image Annotation comes into the picture. Image Annotation helps in labeling the images as per the situation situated in the images, but this is a time consuming task if image data is large in number this is the one disadvantage of labelImg library of python. BERT method can help in this situation. BERT method is Bidirectional Encoder Representations from Transformers and it is created and published in 2018 by the Google. BERT is used for all types of data like text data, image data, video data, etc. But this method is very complex comparative to other neural network model.

CNN model predicts data correctly with 93.34% time with 23.37% loss of features/parameters of the data.

References

[1] Shine L, Jiji CV. Automated detection of helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN.Multimedia Tools and Applications. 2020 Feb 12:1-21.

[2] Espinosa JE, Velastin SA, Branch JW. Motorcycle detection and classification in urban Scenarios using a model based on Faster R-CNN.

[3] Kim H, Park S, Paik J. Pre-Activated 3D CNN and Feature Pyramid Network for Traffic Accident Detection. In2020 IEEE International Conference on Consumer Electronics (ICCE) 2020 Jan 4 (pp. 1-3). IEEE.

[4] Espinosa JE, Velastín SA, Branch JW. Detection of motorcycles in urban traffic using video analysis: A review. IEEE Transactions on Intelligent Transportation Systems. 2020 Jun 9.

[5] Janet, B. and Reddy, U.S, August. Real Time Detection of driver distraction using CNN. In *2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT)* (pp. 185-191) 2020. IEEE.

[6] Diwan A, Gupta V, Chadha C, Diwan A, Gupta V, Chadha C, R-CNN AD. Accident Detection Using Mask R-CNN.

[7] Chirra VR, ReddyUyyala S, Kolli VK. Deep CNN: A Machine Learning Approach for Driver Drowsiness Detection Based on Eye State. Rev. d'Intelligence Artif.. 2019 Dec;33(6):461-6.

[8] Sharma R, Sungheetha A. An Efficient Dimension Reduction based Fusion of CNN and SVM Model for Detection of Abnormal Incident in Video Surveillance. Journal of Soft Computing Paradigm (JSCP). 2021;3(02):55-69.

[9] Madigan R, Nordhoff S, Fox C, Amini RE, Louw T, Wilbrink M, Schieben A, Merat N. Understanding interactions between Automated Road Transport Systems and other road users: A video analysis. Transportation research part F: traffic psychology and behaviour. 2019 Oct 1;66:196-213.

[10] Le TN, Ono S, Sugimoto A, Kawasaki H. Attention r-cnn for accident detection. In2020 IEEE Intelligent Vehicles Symposium (IV) 2020 (pp. 313-320). IEEE.

[11] Yu R, Ai H, Gao Z. Identifying High Risk Driving Scenarios Utilizing a CNN-LSTM Analysis Approach. In2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC) 2020 Sep 20 (pp. 1-6). IEEE.

[12] Shine L, Jiji CV. Automated detection of helmet on motorcyclists from traffic surveillance videos: a comparative analysis using hand-crafted features and CNN. Multimedia Tools and Applications. 2020 Feb 12:1-21.

[13] Kose N, Kopuklu O, Unnervik A, Rigoll G. Real-time driver state monitoring using a cnn based spatio-temporal approach. In2019 IEEE Intelligent Transportation Systems Conference (ITSC) 2019 Oct 27 (pp. 3236-3242). IEEE.

[14] Wan S, Xu X, Wang T, Gu Z. An intelligent video analysis method for abnormal event detection in intelligent transportation systems. IEEE Transactions on Intelligent Transportation Systems. 2020 Sep 9.

[15] Alzraiee H, Leal Ruiz A, Sprotte R. Detecting of Pavement Marking Defects Using Faster R-CNN. Journal of Performance of Constructed Facilities. 2021 Aug 1;35(4):04021035.

[16] D'cunha R, Francis B, Britto R, Shaikh P. Campus Traffic Reinforcement using IoT and OpenCV. In2019 International Conference on Advances in Computing, Communication and Control (ICAC3) 2019 Dec 20 (pp. 1-6). IEEE.

[17] Rodin I, Furnari A, Mavroeidis D, Farinella GM. Scene Understanding and Interaction Anticipation from First Person Vision. InSMARTPHIL-First Workshop on Personal Health Interfaces, in conjunction with ACM IUI 2020.

[18] Nasernejad P, Sayed T, Alsaleh R. Modeling pedestrian behavior in pedestrian-vehicle near misses: A continuous Gaussian Process Inverse Reinforcement Learning (GP-IRL) approach. Accident Analysis & Prevention. 2021 Oct 1;161:106355.

[19] Shen X, Raksincharoensak P. Pedestrian-Aware Statistical Risk Assessment. IEEE Transactions on Intelligent Transportation Systems. 2021 Jun 21.

[20] Al-Mahameed FJ, Qin X, Schneider RJ, Shaon MR. Analyzing pedestrian and bicyclist crashes at the corridor level: structural equation modeling approach. Transportation research record. 2019 Jul;2673(7):308-18.

[21] Chauhan MS, Singh A, Khemka M, Prateek A, Sen R. Embedded CNN based vehicle classification and counting in non-laned road traffic. InProceedings of the Tenth International Conference on Information and Communication Technologies and Development 2019 Jan 4 (pp. 1-11).