ISSN(Online) : *2456-8910*
International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE)
Volume 3, Issue 1, DOI: 10.29027/IJIRASE.v3.i11.2019, 416-419 , July 2019

Pedestrian and Lane Detection using Computer Vision

R. Srija
Department of Computer Science and Engineering
Government College of Technology
Coimbatore, India
Email: srijarajagopal@gmail.com

Dr.K.Kumar M.E.,Ph.D.,
Department of Computer Science and Engineering
Government College of Technology
Coimbatore, India
Email: kkumar@gct.ac.in

***Abstract*: The road network of India is over 52,31,922 kms. During 2015 , the total number of traffic accidents were 4,96,762 with 4,86,567 persons were injured and 1,77,423 were killed.In order to avoid such road accidents, the collision of pedestrian with the vehicle on the highway should be prevented. So an approach for automatic lane and pedestrian detection on highways to prevent road accidents using deep learning and computer vision techniques is proposed. The pedestrian detection is done using tensorflow and HOG detector. Lane detection ensures that the vehicles are within the lane constraints and avoids the collision with vehicles on the nearby lanes.**

Keywords: Computer Vision, HOG detector, Lane detection, pedestrian detection

I.INTRODUCTION

During 2015,97.3% of total traffic accidents took place in 53 mega cities. Out of which, 9.3% of the total road accidents were reported in Chennai. During 2015, maximum fatal road accidents by two wheelers are 43,540 deaths, trucks/lorries are 28,910 deaths, cars are 18,506 deaths and buses are 12,408 deaths. Majority of deaths due to two wheelers accidents were reported in TamilNadu. Inorder to avoid these fatal accidents, Pedestrian and Lane detection is proposed. Pedestrians are detected using tensorflow and HOG detector.Lane detection is done to ensure that the vehicle follows the lane assigned to it and does not collide or interfere with the vehicles in the other lanes.

II. PEDESTRIAN DETECTION

A. HOG Detector

HOG stands for Histograms of Oriented Gradients. HOG is a type of "feature descriptor" whose motive is to generalize the object in such a way that the same object produces as close as possible to the same feature descriptor. Fig. 1 shows the detection of pedestrians in a real time video using HOG detector. The pedestrians are detected using the bounding boxes in Fig. 1. Fig. 2 shows the elapsed time obtained for the detections.

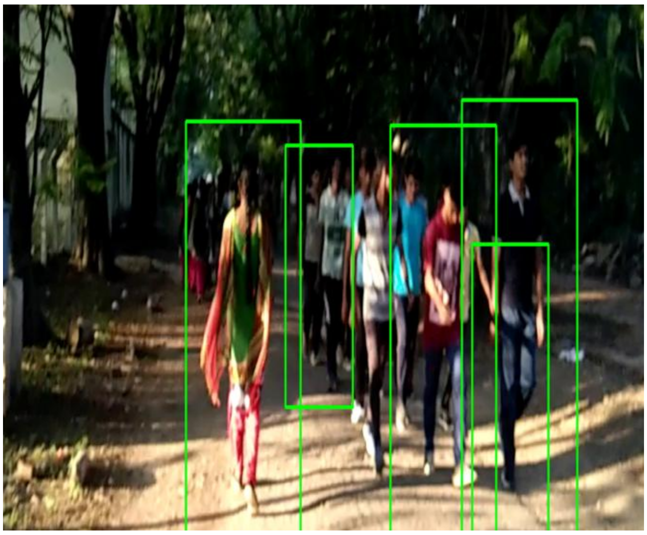


Fig. 1. Pedestrian detection using HOG detector in a real time video

So the task of classification is made easier. Support Vector Machine is used to recognize HOG descriptors of people. The HOG person detector uses a sliding detection window Technique.

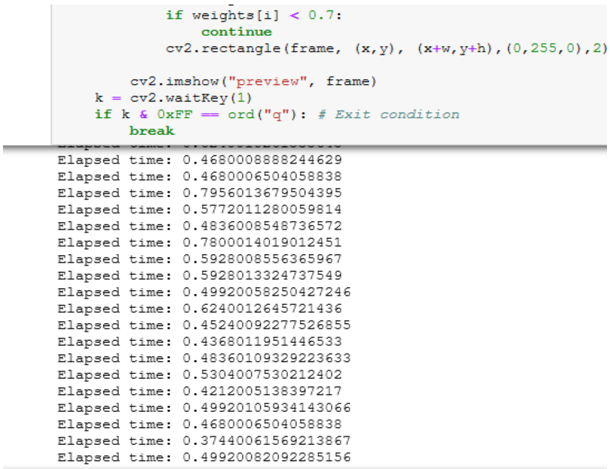


Fig. 2. Measure of Elapsed time for detections

ISSN(Online) : 2456-8910
International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE)
Volume 3, Issue 1, DOI: 10.29027/IJIRASE.v3.i11.2019, 416-419 , July 2019

The window is moved around the image. HOG descriptor is computed at each position of the detector window. Then the SVM classifies it as either "person" or "not a person". To recognize persons at different scales, the image is resized to multiple sizes. The HOG person detector uses a detection window that is 64 pixels wide by 128 pixels tall.HOG needs a grayscale image.To generate detection boxes ,the HOG detector returns slightly larger rectangles than the real objects. so we slightly shrink the rectangles to get a better output.The captured real time video is given as the input. Elapsed time is calculated for the detections. The detection boxes are drawn using the coordinates in the frames.

B. Object Detection using Tensorflow

Tensorflow uses Deep Learning for computation**.** The TensorFlow Object Detection API framework enables to construct, train and deploy object detection. This API detects objects in images or video using bounding boxes. Detection, Recognition and localization of multiple objects within an image is done. The workflow of object detection involves Feature extraction, Classification and Testing. Feature extraction extracts features from the input images. Using the extracted features, the class of the image is determined.

Initially the necessary packages are installed. COCO dataset is used for detection. So the model which is trained on this dataset is downloaded. COCO dataset contains around 330K labeled images. For training the detection model, width and height are required for each image .The bounding box is the frame that captures the class in the image. A real time video is given as input. Processing a video file will take three steps

1. Convert the video to images frames

2. Process the images

3. Convert the processed images to output videos

The output consists of the detection of objects with labels and detection scores of that object being similar to the training data. Fig. 3 shows the detection of pedestrians in a real time video using tensorflow with labels and detection scores.

METHODOLOGY FOR OBJECT DETECTION

Import the necessary packages

Packages such as numpy,scipy,opencv,Matplotlib, MoviePy,Tensorflow are imported.These packages are necessary for detection in a video.

Collect Images

COCO Dataset is a large-scale dataset used for object detection, segmentation, and captioning. It is used to provide the images necessary for detection.

Generate Annotation

Annotation is the process of labelling data in the form of text. The annotated data is used to recognize similar patterns in a new data.

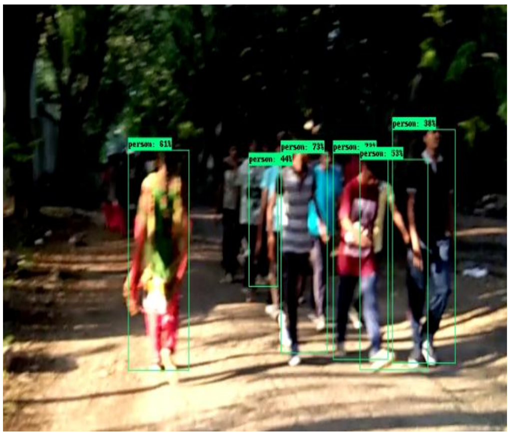


Fig. 3. Pedestrian detection using tensorflow with labels and detection scores

Training Data

It provides an algorithm to learn from the training data. The output is the model that captures these patterns.

Testing

Test the model on the real time video. The video is parsed into frames and the object detection is done on each frame. Detection boxes and detection scores are generated.

III. LANE DETECTION

When driving a vehicle, the lanes acts as the reference lines. The lanes guide the driver to steer the vehicle in the right direction. Lane detection plays a crucial role in ensuring that the vehicles are within the lane constraints and avoids the collision with vehicles on the nearby lanes. The lane lines are detected using computer vision algorithms. Fig. 4 shows the detection of the appropriate lane. Fig. 5 shows the total time taken for detection of lane.

Fig. 4. Detection of the appropriate lane

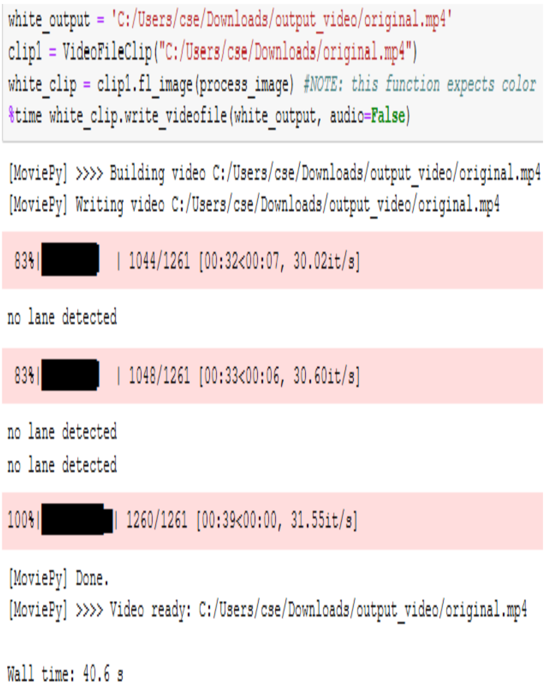
ISSN(Online) : 2456-8910
International Journal of Innovative Research in Applied Sciences and Engineering (IJIRASE)
Volume 3, Issue 1, DOI: 10.29027/IJIRASE.v3.i11.2019, 416-419 , July 2019

B. *Methodology for lane detection*

Convert frame to grayscale. So the RGB frames are to be converted to grayscale. The color of the lane lines are yellow and white. Masks are created for yellow and white pixels. Gaussian smoothing is applied. This filter suppresses the noise by computing the average of the neighboring pixels. In the next step, the Canny edge detection is applied. It is used to parse the pixel values according to their directional derivative or gradient. The driver need not focus on the vehicles in the other lane. So an additional mask is created to focus on the "region of interest" in front of the vehicle. To work only with the relevant edges, set everything outside of the ROI to black/zero. The slope is calculated using the formula in (1).

$$slope=(y2-y1)/(x2-x1) \qquad (1)$$

The pixels in XY space are converted to a line in Hough space. A line is said to exist in XY space ,where the lines in Hough space intersect. Each individual line returned by hough is examined to determine if the line is in left or right lane using its slope. Using the extrema of the lines generated, two averaged lines across frames for a smooth video playback is created. Draw the lines to each frame. A real time video taken in a lane is given as the input. The input video is processed to detect the appropriate lane within which the vehicle is moving.



```
white_output = 'C:/Users/cse/Downloads/output_video/original.mp4'
clip1 = VideoFileClip("C:/Users/cse/Downloads/original.mp4")
white_clip = clip1.fl_image(process_image) #NOTE: this function expects color
%time white_clip.write_videofile(white_output, audio=False)

[MoviePy] >>>> Building video C:/Users/cse/Downloads/output_video/original.mp4
[MoviePy] Writing video C:/Users/cse/Downloads/output_video/original.mp4

83%|          | 1044/1261 [00:32<00:07, 30.02it/s]

no lane detected

83%|          | 1048/1261 [00:33<00:06, 30.60it/s]

no lane detected
no lane detected

100%|          | 1260/1261 [00:39<00:00, 31.55it/s]

[MoviePy] Done.
[MoviePy] >>>> Video ready: C:/Users/cse/Downloads/output_video/original.mp4

Wall time: 40.6 s
```

Fig. 5. Total time taken for detection of lane

IV. CONCLUSION

The detection of pedestrians using the HOG detector detects pedestrians with detection boxes. *In object detection using* tensorflow, the objects are detected using the bounding boxes with detection scores and labels. Lane detection ensures that the vehicle travels in its appropriate lane. It does not interfere with vehicles on the other lanes. Using the approach for automatic lane and pedestrian detection on highways using computer vision techniques, road accidents can be prevented and fatal issues can be avoided.

REFERENCES

[1] Sachin Umesh Sharma and Dharmesh J. Shah, “A Practical Animal Detection and Collision Avoidance System Using Computer Vision Technique.”, IEEE Translations and content mining,Vol.5,pp.347 - 358,February 2017.

[2] Weicheng Sun ,Songhao Zhu, Xeuwen Ju, Dongsheng Wang, “Deep Learning Based Pedestrian Detection” ,The 30th Chinese Control and Decision Conference,pp.1007-1010, July 2018 .

[3] Qichang Hu, Peng Wang, Chunhua Shen, Anton van den Hengel, Fatih Porikli, “Pushing the Limits of Deep CNNs for Pedestrian Detection” ,IEEE Transactions on Circuits and Systems for Video Technology, Vol.28,pp.1358 - 1368 ,June 2018. I.S. Jacobs and C.P. Bean, “Fine particles, thin films and exchange anisotropy,” in *Magnetism*, vol. III, G.T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271-350.

[4] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik ,”Region-based Convolutional Networks for Accurate Object Detection and Segmentation”,IEEE Transactions on Pattern Analysis and Machine Intelligence Year: 2016, Vol: 38,pp.142 - 158 ,Jan 2016.

[5] Xiaotong Zhao, Wei Li, Yifan Zhang, T. Aaron Gulliver, Shuo Chang, Zhiyong Feng, ”A Faster RCNN-based Pedestrian Detection System”, 2016 IEEE 84th Vehicular Technology Conference ,March 2017.

[6] Sachin Sharma, Dharmesh Shah , “Real-Time Automatic Obstacle Detection and Alert System for Driver Assistance on Indian Roads”,Indonesian Journal of Electrical Engineering and Computer Science,vol. 1, no. 3, pp. 635- 646, March 2016.

[7] Jittima Varagulaa, Panit-a-nong Kulpromaa, Toshio Itobb ,“Object Detection Method in Traffic by On-Board Computer Vision with Time Delay Neural Network”,International Conference on Knowledge Based

and Intelligent Information and Engineering Systems,vol.no.112,pp.127-136,September 2017.

[8] Yeong-Hyeon Byeon,Keun-Chang Kwak, "A Performance Comparison of Pedestrian Detection Using Faster RCNN and ACF",2017 6th IIAI International Congress on Advanced Applied Informatics (IIAIAAI),pp.858-853, July 2017.

[9] Carlos Ismael Orozco, Maria Elena Buemi,Julio Jacobo Berlles,"New Deep Convolutional Neural Network Architecture for Pedestrian Detection",8th International Conference of Pattern Recognition Systems ,July 2017.

[10] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks" ,IEEE Transactions on Pattern Analysis and Machine Intelligence ,Jan 2016.

[11] Kaiming He ,Georgia Gkioxari ,Piotr Dollar ,Ross Girshick, "Mask R-CNN" ,2017 IEEE International Conference on Computer Vision ,March 2017.

[12] http://ncrb.gov.in/