Bounding Box Alignment Based Pedestrian Crossing Collision Avoidance Using Convolution Neural Networks

Arunkumar E.
and
Sunilkumar S. Manvi
School of Computing and IT, REVA University, Bengaluru, India

Pedestrian detection is a challenging task for autonomous vehicles in an urban environment. Pedestrian in videos has a Variety of appearances such as occlusion and body poses and there is a proposal shift problem in pedestrian detection that cause the loss of parts such as legs and head. To address such a problem, we suggest part-level convolution neural networks based method for pedestrian recognition using saliency map and boundary box framework in this paper. The proposed method consists of two sub-networks: person-detection and alignment. We use saliency map along with weights in the detection sub-network to remove false detections such as lamp posts and trees. The alignment network employs confidence map for better prediction of pedestrian alignment. The method is implemented and analyzed on various data sets and it hass been observed that the proposed method has better accuracy and low false positives than the existing methods.

Keywords: Occlusion handling, false positive removal, convolutional neural network, Pedestrian detection, Boundary box alignment, Saliency

1. INTRODUCTION

Intelligent Transportation System (ITS) applies advanced technologies of electronics, communications, computers, control, sensing and detecting in all kinds of transportation system in order to improve safety, efficiency and service, and traffic situation through transmitting real-time information. Autonomous vehicles or self driving cars have numerous problems. Detecting a human on the roads is one of the important problems which have great importance in the field of computer vision and autonomous vehicles. For multiple purposes, from video surveillance systems to cars (vehicles) with automatic pilot, there have been continuous improvements in the techniques for detecting humans.

Intelligent transportation system consists of the following:

a) Advanced Vehicle Control and Safety System (AVCSS)

b) Advanced Traveler Information System (ATIS)

c) Advanced Public Transportation System (APTS)

d) Commercial Vehicle Operation (CVO)

AVCSS applies new technologies in roads and vehicles, and helps control vehicles in order to minimize the accidents and improve road safety. The AVCSS mainly includes pedestrian crossing collision avoidance, driving assistance, automatic longitudinal control, and automatic driving and automatic highway system. We have considered AVCSS in this paper focusing on pedestrian collision detection problem.

In [Felzenszwalb and McAllester 2008] authors proposed a star model [Jie et al. 2017] [Chen et al. 2015] to search the whole image for body parts by a multi-scale sliding window technique. This work has inspired researchers to consider part detection in deep learning designed with a unique part detection layer comprising 20 convolutional filters of different sizes to detect body parts of the corresponding size ratio. There are several issues in pedestrian detection methods which are as follows. 1) Detection of occluded pedestrians, 2) Avoiding of false detections, 3)

Detecting pedestrians with complex background, 4) Detection of overlapped pedestrians, and 5) Processing of images taken at night or low lighting conditions.

In this paper, we propose part-level convolution neural networks (CNN) for pedestrian detection using fully convolutional networks (FCN). The proposed network consists of two sub-networks: detection and alignment. In detection sub-network, we use saliency to assign different weights to pedestrians and background. Based on saliency, we remove false positives such as lamp posts and trees from pedestrians through occlusion handling. We adopt the alignment sub-network to recall the lost body parts caused by the detection sub-network. In alignment sub-network, we utilize localization features of CNN such as FCN and CAM (class activation map) to produce confidence maps and infer accurate pedestrian location, i.e. bounding box alignment.

Our contributions in this paper are as follows.

(1) Saliency maps are used for occlusion handling with reduced number of layers
(2) Confidence maps are employed to reduce number of false positives per image
(3) The method is analyzed for accuracy and false positives
(4) Proposed method is better as compared to existing methods.

The rest of this paper is organized as follows. Section 2 presents relevant research trends. In Section 3, the proposed method is described in detail. Sections 4 and 5 describe experiments and results, respectively. Section 6 concludes the work with future enhancements.

2. RELATED WORK

Researchers have proposed many outstanding works for pedestrian detection, and in this section we mainly focus on deep learning models for pedestrian detection. In [Felzenszwalb and McAllester 2008], authors adopt deep learning to deal with the proposal shifting problem in pedestrian detection. Firstly, CNN and FCN are combined to align bounding boxes for pedestrians. Secondly, part-level pedestrian detection based on CNN to recall the lost body parts is performed. ConvNet [Nam et al. 2017] is typically pre-trained with massive general object categories (e.g. ImageNet). Although these features are able to handle variations such as poses, viewpoints, and lightings, they may fail when pedestrian images with complex occlusions are present. Unlike previous deep models that directly learned a single detector for pedestrian detection, DeepParts [Shaoqing and Kaiming 2016] [Piotr Dolla et al. 2016] [Carlos and Maria 2017] has several appealing properties which leverage the RPN (Region proposed network) architecture of Faster R-CNN and extend it to a multi-layer version combined with skip pooling to tackle the pedestrian detection problem. Skip pooling is a kind of network connection that combines multiple ROI (Region of interest) pooling results from lower layers to form a single input to a higher layer while bypassing intermediate layers.

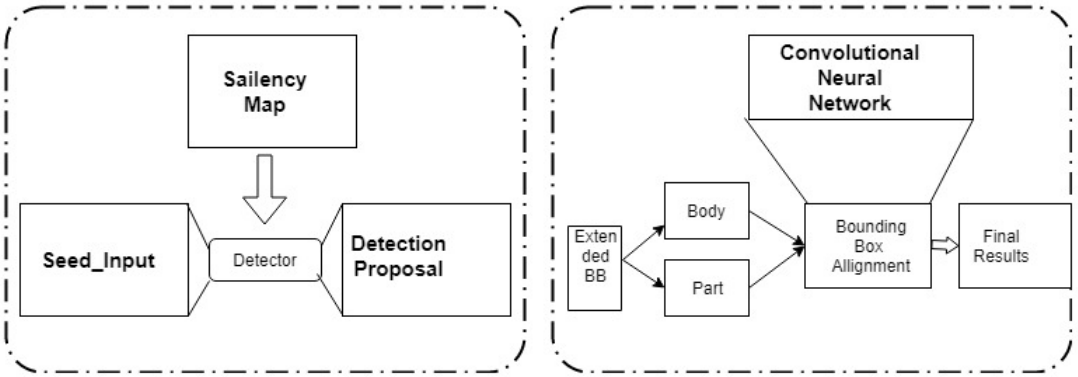
Authors in [Wang and Jung 2010] comprehensively evaluate network, referred to as SP-CNN, on the Caltech pedestrian detection benchmark and KITTI object detection benchmark. Advances like SPPnet [Felzenszwalb and McAllester 2008] and Fast R-CNN [Felzenszwalb and Gieshick 2017] have reduced the running time of these detection networks, exposing region proposal computation as a bottleneck. Region Proposal Network (RPN) is introduced that shares full-image convolutional features with the detection network, thus enabling nearly cost-free region proposals. A RPN is a fully convolutional network that simultaneously predicts object bound scores at each position. The RPN is trained end-to-end to generate high-quality region proposals, which are used by Fast R-CNN for detection [Xinchuan et al. 2018]. Authors in [Yonglong and Ping 2016] proposed effectiveness of self-similarity features and compared with deep learning experiments to show that deep learning has poor performance in terms of accuracy and detection speed. Authors in [10, 13] improved HOG (Histograms of Oriented Gradients) to track pedestrian in real time. Authors in [Datasets] have developed deep convolutional neural network architecture for detection of pedestrians to evaluate pedestrian and non pedestrians using pyramidal sliding window approach. Deep learning methods have achieved great successes in pedestrian detection,

owing to its ability to learn discriminative features from raw pixels. However, they treat pedestrian detection as a single binary classification task, which may increase the false positives and reduced the accuracy.

3. PROPOSED METHOD

The proposed pedestrian detection method has two approaches one is detection of pedestrian and another is an alignment, i.e. bounding box alignment for pedestrians. Figure 1 illustrates the framework of the proposed method. The model mainly included two sub-networks: detection and alignment. These two perform the following.

(1) Detection of pedestrian with body parts using saliency map by using visualized parts of input
(2) Alignment of the obtained body parts by using convolutional neural network (CNN) for better prediction of outputs



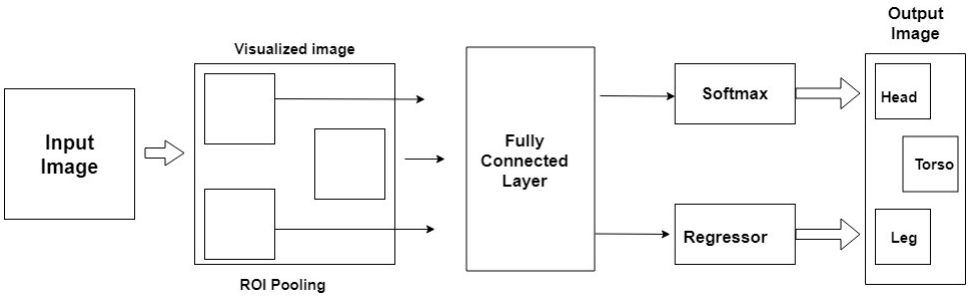
. Fig. 1: Framework of proposed method

3.1 Detection Framework

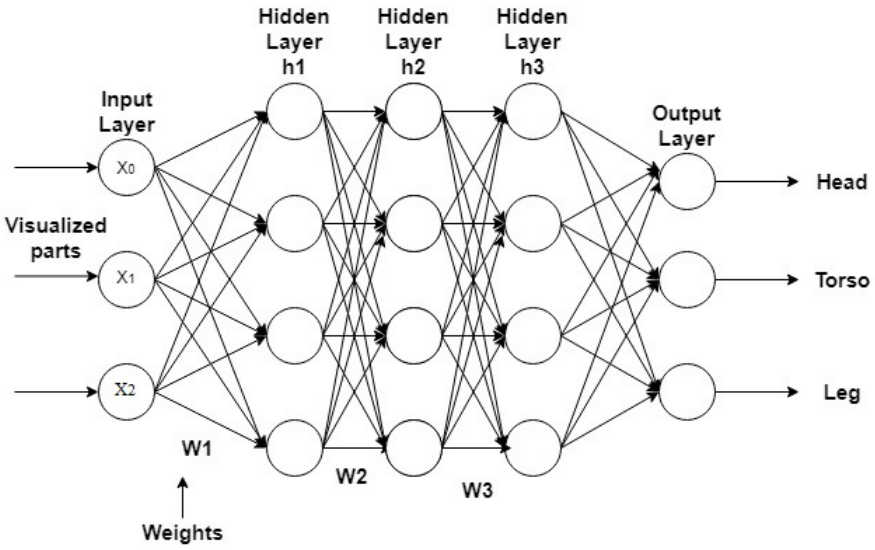
Saliency is an important concept used in detection which uses homogeneity pixels for detection. It is explained as follows. The area (pedestrian) based division in saliency is partitioning of a picture into comparative/homogenous areas of associated pixels through applying of homogeneity standards among competitor units of pixels. Every one of the pixels in every place has similarities with relevancy (body parts) and few characteristics (rest of covered body parts) or computed property like shade, depth and/or texture. Failure to regulate the homogeneity/similarity criteria consequently can turn out undesirable results.

Figure 2 illustrates the architecture of detection framework which comprises of input image, visualized image constructed through global max pooling for handling saliency (minimize/maximize), fully connected (FC) layer of CNN, softmax of CNN for normalization of output from FC layer, regressor for depth prediction, and classified image (head, torso and leg). We use faster R-CNN [3] to obtain the detection proposals for pedestrians. Least number of hidden layers are used, i.e. 3, which are for head, torso and leg. Thus inputs are reduced. The detection results include some false positives such as lamp post, trees and vehicles parts. For removing false positives, we applied different weights in FC layer, thus detection algorithm focuses mainly on the pedestrians[Felzenszwalb and McAllester 2008].

The FC layer is depicted in Figure 3. It has three inputs (an object is divided into three parts - visualized parts of input) and twelve hidden units and three classified outputs (head, torso, leg). Three hidden layers (neurons) are interconnected with weights. The outputs of FC hidden layers is given in Eq. 1.



. Fig. 2: Architecture of detection framework



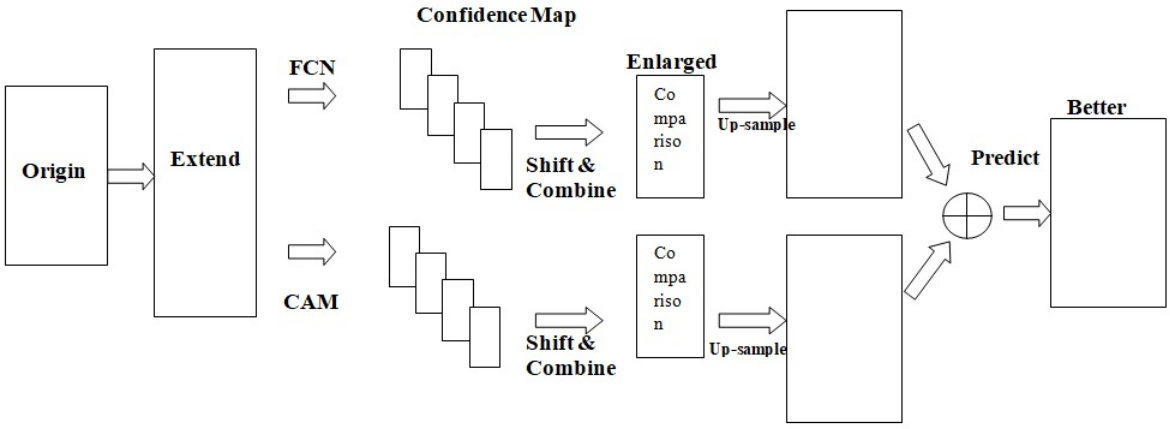
. Fig. 3: Internal architecture of fully connected layer

$$Y = \sum_{i=1}^{x} X_i W_i \tag{1}$$

Where Xi is number of inputs, Wi is Weights Applied, Li is number of hidden layers.

3.2 Alignment Framework

The second stage is an alignment framework using part-level detection. Our part-level detection is a combination of one root detector and three root detectors. One root detector detects root position of pedestrians and three root detectors detect human body parts such as torso, head and legs. Theoretically, bounding box alignment helps the proposed detector by better detection proposals as well as recalls the lost body parts which are out of the ground truth. The alignment network generates a weighted map of the confidence scores with a spatial distance penalty term as the final confidence score of a detection proposal as depicted in Fig.4. The nomenclatures used in the figure are as follows. Origin: Original bounding box. The pedestrian is localized at the top left corner of a bounding box. Extend: Enlarged bounding box. Confidence map: Output of FCN (Fully Connected Network) and CAM (Class Activation Map). Better: Aligned bounding box - the lost head part is recalled and thus the pedestrian is accurately localized.



. Fig. 4: Pipeline for bounding box alignment

The pipeline of a bounding box alignment works as follows.

(1) Original bounding box of the detected image is considered
(2) Extend the detected original image to cover the various parts of body
(3) Generate confidence map by using FCN and CAM
(4) Combine the confidence maps of FCN and CAM, respectively
(5) Compare the original image with image from confidence map
(6) Up sample the original image to obtain complete image of the pedestrian
(7) Combine up sampled images to obtain better prediction

Bounding box will be improved by applying weights to FCN. To determine the weight of the bounding box, we obtain pedestrian saliency maps using some of the saliency network from input image which is known as seed image. Using Eq.2, updating of class probability for saliency maps takes place.

$$fw(b) = f(b) * wf \tag{2}$$

The weight wf is defined as follows:

$$Wf = \begin{cases} 1\text{if } f(b) > th_b \\ \frac{1}{N}\sum_x, y \in bs(x,y)\text{otherwise} \end{cases}$$

Where b is bounding boxes of proposals, $s(x, y)$ is saliency score in the position (x, y), and $f(b)$ is class scores of the selected bounding box. th_b is the threshold value.

3.3 Functioning of Method

The overall functioning of method is as follows. Saliency algorithm detects the pedestrian in a sequence of video and the area where pedestrian exists. It ignores the least number of false positive and color similarities which often look like a pedestrian. Next, alignment network obtains

confidence maps for both FCN and CAM which is combined together for betterment of pedestrian detection. The certain advantages of this proposed method are s follows. Saliency mapping concept of pedestrian detection is capable of detecting occluded pedestrian and remove some of false positive per images. Number of hidden layers in the network is reduced to 3, and employs 3 numbers of inputs. The disadvantage of the work is that it used weights (from trained data file) for false positives because of some complex color similarities. Thus, there are less wrong detection.

4. EXPERIMENT

In this section, we present datasets, tools, inputs and performance parameters considered in experiment.

4.1 Dataset

As shown in Table 1, list of different positive and negative datasets are considered for training and testing[Datasets]. Dataset consists of approximately 10 hours of 640*480 30Hz video taken from a vehicle driving through regular traffic in an urban environment. About 250,000 frames are present in 10 hours of video. We use every frame to train and considered every frames output.

4.2 Tools

We have implemented the entire learning network using Python Language because of its interpretation. It is high level programming language helps programmers to write clear a logic code for projects. To work with python language there are different platforms such as anaconda, pycharm, IDLE etc. we have considered anaconda because its distribution is used by over 13 million users and includes more than 1400 popular data-science packages suitable for Windows, Linux, and MacOS. We consider jupyter notebook for python programming with different packages installed, mainly Tensorflow [Ouyang and Wang 2012], keras and CV2. We have implemented the proposed work on a Laptop with Intel-i3 processor, 4GB RAM and 500GB memory.

4.3 Inputs

We consider the different datasets such as Caltech, INRIA, and ETH. These are used to train the number of negative and positive images as listed in Table 1 in order to remove the occlusion and false positives. Video data is obtained in the form of sequence data (.seq file) and considered as input.

4.4 Performance Parameters

Parameters evaluated in proposed work and compared works are accuracy and FPFI (false positive per image). Accuracy for Precision is a description of random errors, a measure of statistical variability. Accuracy is measured over Epoch. Epoch for UNIX time is a system for describing a point in time. FPFI for finding the number of false positive detection or appearance per image is the percentage of accesses from different methods as given Eq.3.

$$FPFI = \frac{area(BB_{dt} \cap Bb_{gt}}{area(BB_{dt} \cup BB_{gt}} \tag{3}$$

Where BB_{dt} and BB_{gt} are detection bounding box and ground truth bounding box, respectively.

5. RESULTS AND DISCUSSION

We conduct a set of experiments on different dataset to investigate the detection accuracy and FPFI of the proposed method and the existing methods [[Felzenszwalb and McAllester 2008], [Wang and Jung 2010], [Nam et al. 2017]].

Datasets	Training			Testing			Height			properties		
	Pedestrian	Negatives	Positives	Pedestrian	Negatives	Positives	10th Percentile	Median	90th Percentile	Colour images	Occluded	Video sequence
Caltech	1832	1218	614	1100	750	350	27	48	90	yes	yes	yes
INRIA	1208	1100	714	566	453	288	20	35	60	yes	No	No
ETH	1288	--	499	1200	---	1804	40	23	80	yes	no	yes

. Table 1: The number of data (positive and negative) considered from different database for training and testing

5.1 Analysis of FPFI

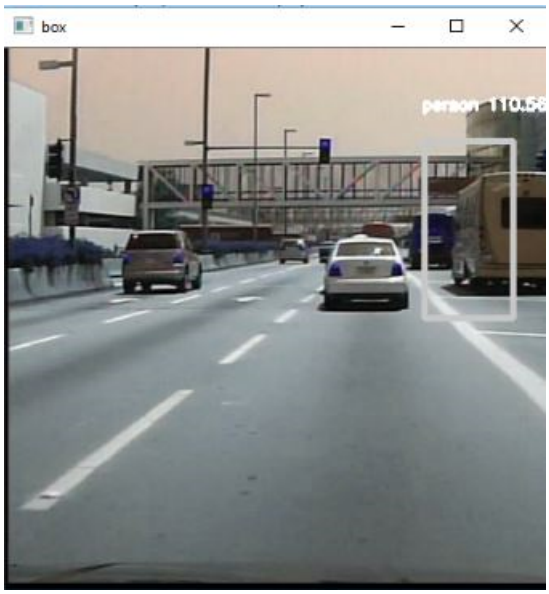
There are total 250 detected and annotated true positive pedestrians in 2000 frames. Rest of the frames is considered as FPFI depending on the color similarity and weights. Table 2 presents comparison of our work with other works. We notice that proposed work has less FPFI, 18.04 percent as compared to other works. The confidence maps will facilitate reduced number false detections.

Methods	FPPI (%)
Joint Deep[1]	39.3
CifarNet[3]	28.4
TA-CNN[7]	20.9
Proposed Work	19.08(as per frame calculation and bounding box alignments)

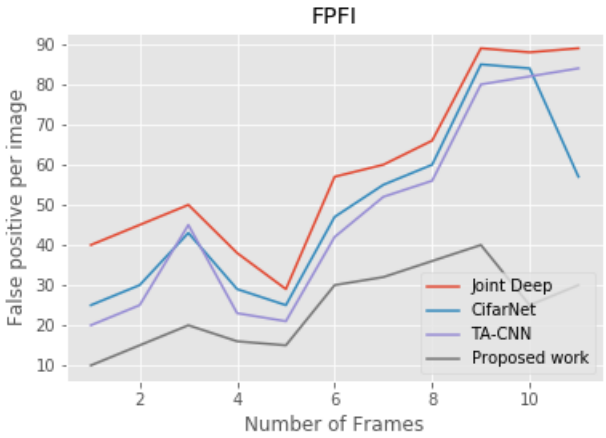
. Table 2:Performance comparison between different methods

Figure 5 shows a snapshot of wrong detection of pedestrian as vehicle part using the proposed method, because of color similarity. We have applied different weights to model reduction of FPFI by removal of detection of trees and lamp post. Proposed method considers only body parts of a pedestrian

We observe from Figure 6 that FPFI is better in proposed method compared to other methods. This is considered for false positive image wise in entire video, i.e. in a sequence of frames. The



. Fig. 5: False positives per image (Vehicle part)



. Fig. 6: Performance comparison of FPFI

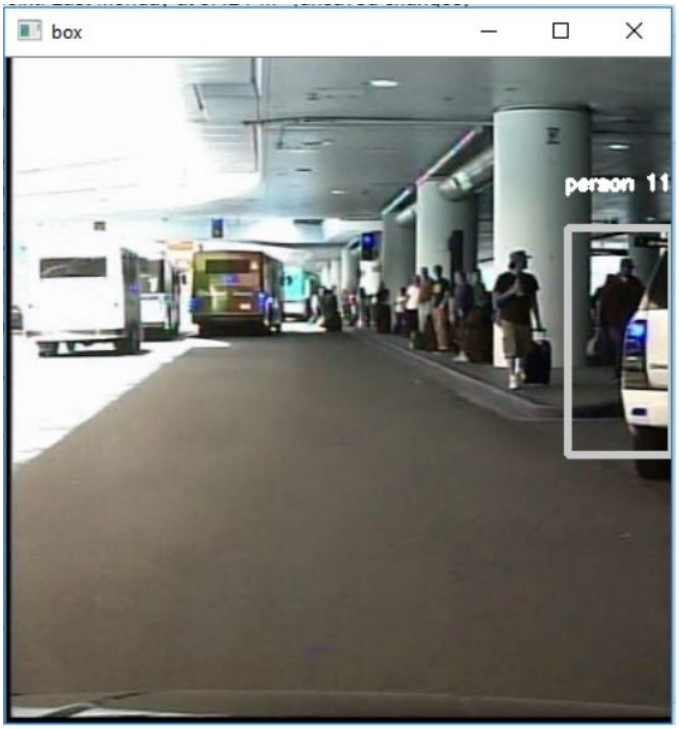
number of false images is decreased. This is due to the fact that the proposed network used weights to remove some of false positives

Figure 7 shows some successful detection results (a and b) - Snapshot of a sample images detected with occlusion. The sample snapshots of the detection with occlusion are shown in Fig. 7. The figures 7(a.1) and 7(b.1) show the detection results of basic (without saliency) and figures 7(a.2) and 7(b.2) show images with proposed (with "Saliency + Part Detectors") work, respectively. These images are true positives.

5.2 Analysis of Accuracy

Here we present accuracy of the proposed work. The other works are not considered for comparison since they do not handle occlusion as they aim at pedestrian detection only. We have reduced the number of hidden layers by using three inputs, head, torso and leg. The method removes the occlusion.

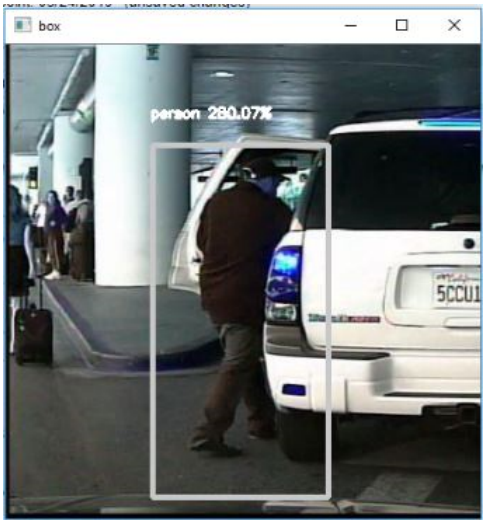
. Fig. 7(a.1): Snapshot without saliency



. Fig. 7(a.2): Snapshot with saliency

Accuracy (in seconds) is depicted in Figure 8 for occluded pedestrian. Training accuracy for detection (after training) varies from 0.2 to 1.0 seconds considering various epochs. Valid accuracy with occlusion detection varies from 0.13 to 0.9 seconds consider various epochs. Proposed method detected the occluded pedestrian within one second. For training positive data sets, method takes 274 seconds. 4861 seconds are taken to train negatives considering 8 values from each frame of

. Fig. 7(b.1): Snapshot without saliency

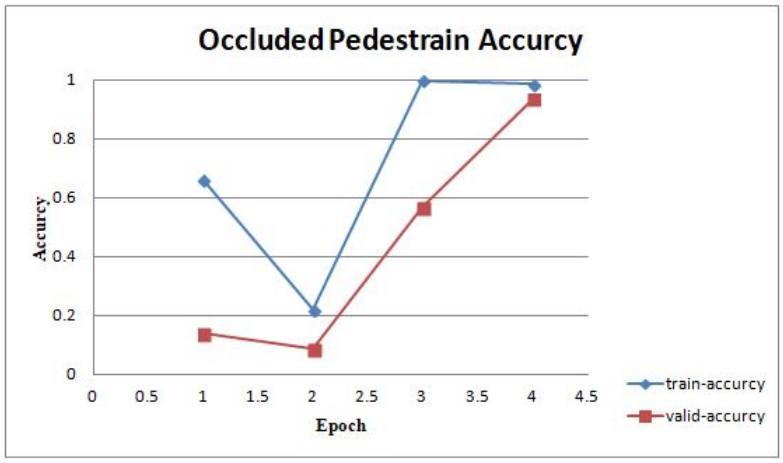


. Fig. 7(b.2): Snapshot with saliency

pedestrian detection, i.e. 8 moves of single pedestrian in a frame. Thus, 30.875 seconds is the time taken for pedestrian detection. For just pedestrian detection, our work does faster detection that the work given in [Felzenszwalb and McAllester 2008] which takes 60.45 seconds.

6. CONCLUSIONS

In this paper, we have proposed part-level CNN for pedestrian detection using saliency and boundary box alignment. We have used saliency in the detection sub-network to remove false positives such as lamp posts and trees. We have utilized boundary box alignment in the alignment sub-network to recall the lost body parts. Experimental results demonstrate that the proposed method achieves competitive performance on Caltech datasets for pedestrian detection in terms of FPFI. In future, we would like to investigate pedestrian detection in low light conditions with the help of near Infrared (NIR) data and devise solutions to process the images to reduce FPFI



. Fig. 8: Accuracy of occlusion pedestrian

and give better accuracy. Furthermore, work can be extended to predict overlapped pedestrian with other objects.

REFERENCES

Carlos, I. and Maria, E. 2017. New deep convolutional neural network architecture for pedestrian detection. Proc.Int.conf, 8th International Conference of Pattern Recognition Systems (ICPRS).

Chen, Q., Wenhui, J., and Yanyun, Z. 2015. Part-based deep network for pedestrian detection in surveillance videos. pp.649–654.

Datasets. Datasets caltech. `http://www.vision.caltech.edu/Image_Datasets/CaltechPedestrians/datasets/USA/`.

Felzenszwalb, P. and Gieshick, R. 2017. Object detection with discriminatively trained part-based models. pp.1627–1645.

Felzenszwalb, P. and McAllester, D. 2008. A discriminatively trained, multiscale, deformable part model. pp.1063–1075.

Jie, L., Xingkun, G., and Nianyuan, B. 2017. Deep convolutional neural networks for pedestrian detection with skip pooling. pp.2161–2170.

Nam, W., Dollar, P., and Han, J. 2017. Local decorrelation for improved pedestrian detection. pp.424–432.

Ouyang, W. and Wang, X. 2012. A discriminative deep model for pedestrian detection with occlusion handling. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition.

Piotr Dolla, R., Christian, W., and S, B. 2016. Pedestrian detection: An evaluation of the state of the art. pp.743–761.

Shaoqing, R. and Kaiming, H. 2016. Faster r-cnn: Towards real-time object detection with region proposal networks. pp.1137–1149.

Wang, X. and Jung, C. 2010. Part-level fully convolutional networks for pedestrian detection. pp.2267–2271.

Xinchuan, F., Rui, Y., and Weinan, Z. 2018. Pedestrian detection by feature selected self-similarity features. pp.18–25.

Yonglong, T. and Ping, L. 2016. Deep learning strong parts for pedestrian detection. pp.1904–1912.

Dr. Sunilkumar S Manvi is presently working as the Director, School of Computing and Information Technology, REVA University, Bengaluru, India. He pursued PhD from IISc Bangalore and has around 30 years of teaching and research experience. He has more than 400 publications in reputed journals and conferences. He is a senior member of IEEE, ACM and life member of CSI. He has executed several sponsored research projects and is a recipient of "Prof. Satish Dhawan Young Scientist Award in Engineering Sciences" in 2015.

Arun Kumar pursued his M.Tech in Computer Science Engineering from REVA University, Bangalore. He is currently working in Capegemini, Bengaluru, India. His areas of interest are machine learning, data science, and computer networks