IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

REAL TIME PEDESTRIAN DETECTION

Prof. Karthikeyini[1], Adarsh AV[2], Akhilesh A[3], Aswin N L[4], Prathin Pratheesh[5]

[1]ASP-CSE, JCT College of Engineering & Technology

[2-5]BE Computer Engineering, JCT College of Engineering & Technology

Abstract: Object detection is the process of determining the presence, location, and type or class of at least one object using a bounding box. The person detection process produces a bounding box and allot a class label as a person based on YOLO v3. In YOLO v3 the features are learned, divides the image cells and each cell says a bounding box and entity classification directly. There could be more than one bounding box per person, but the system makes use of non-maximum suppression to reduce the number of bounding boxes to one per person. Finally, the number of persons in the image and video are calculated using the count of the bounding boxes. The dataset used for static pedestrian detection is the INRIA dataset and ShanghaiTech dataset. Yolo_Mark is used for marking bounding boxes of persons and gets its annotation files using 243 images from the INRIA dataset. Darknet is used as the framework for implementing YOLOv3. From INRIA Dataset 120 images are used for testing purposes. Testing on the INRIA dataset resulted in an accuracy of 96.1%. From the Shanghai tech-B, dataset 56 images are used for testing. Testing resulted in an accuracy of 87.3%.

Keywords: Yolo, CNN, CUDA.

I. INTRODUCTION

Computer vision is the region of study that expects to build up a technique that upholds computers to watch and comprehend the substance of advanced pictures, for example, photos and recordings. Computer vision has a number of multidisciplinary applications like military human-computer interaction, mobile robot navigation, industrial inspection, and medical image analysis. Many popular computer vision applications like object classification, object identification, object verification, object detection [3], and object recognition recognize objects in images or videos.

Object detection has two levels; Object tracking and Object localization. Object tracking is the process to detect and track individual objects and grouped objects.Object Detection is made easy with the introduction of Neural Network. Neural networks are used to simulate the human cerebrum framework to take care of general learning issues. Neural networks use numerous algorithms that help to identify patterns. Deep learning [1] is the group of machine learning networks. Deep learning has become a popular implementation of speech recognition [2]. Deep learning is the field of computer vision, which contains better performing models that may require more data but less digital signal processing expertise to train and operate.

A CNN, or ConvNet [4] is gaining much popularity in the area of deep learning. A CNN has an input, multiple hidden and an output layer.The pooling layer in the form of non- linear down-sampling and the most commonly used one is max pooling function. Finally, the fully connected layers done high-level reasoning.YOLO is the second group of strategies object recognition focused on real-time processing. YOLO represents You Only Look Once, introduced in 2015.

Outdoor pedestrian counting techniques help to monitor parks, recreational facilities, and trails help to recognize and count the number of individuals visiting public spots. The system assists in designing traffic infrastructure according to the peak traffic experienced in an area and finding out the bottlenecks in the existing traffic system. It also helps in calculating the capacity ratings for the traffic infrastructure like bridges while designing the infrastructure projects. During pedestrian detection, there might be many bounding boxes representing different entities of regard within the image, and it is difficult to know how many beforehand.

ILSVRC is a strategy including classification and recognition of several object classes and images. Image classification process algorithms generate a catalogue of entity classes there in the image, with a bounding box representing the location[5]. The R-CNN, Regions with CNN Features was presented by Ross Girshick et al. in the year of 2014. Selective search [6] choosing a vast number of regions using this method to get only 2000 regions.

The R-CNN model [7], comprises three modules: Region Proposal, Feature Extractor, and Classifier. AlexNet deep

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

CNN [8] is the feature extractor used here. Linear SVM is theclassifier that takes the output of the CNN layer.

To solve the speed issue of R-CNN, Ross Girshick, introduced an extension to create a faster object detection method, Fast R-CNN [9] in 2015. This technique is similar to the R-CNN process. This model took the image and a group of region recommendations. For feature extraction, a pre- trained CNN using VGG-1 is employed. To improve the speed of training, detection techniques, Faster R-CNN [10] inthe year 2016.

A. Motivation

All the advanced methods use locales to detect the object within the image. Those methods only consider the image pieces that have peak probabilities of acquiring the object. YOLO applies a single neural network trained process that takes an input image and plots bounding boxes for entities with corresponding class labels. YOLO looks at the image once and makes predictions with a single network evaluationmodel, which makes YOLO fast. YOLO is trained on full images and directly optimized detection performance. For improving the accuracy, YOLO reached out to variants YOLO v2 and YOLO v3.

B. Contributions

The proposed method contributes an efficient pedestrian detection and counting using the YOLO v3 model. The proposed method detects the person on both images and videos. It handles images of different lighting conditions, varying viewing angles, and scales. Estimate the count of detected persons in the images, using the Inria dataset and Shanghaitech-B dataset. Estimate the peak count of persons in the videos.

II. RELATED TECHNIQUES

In [11], Zhang et al. introduced a technique that can calculatethe crowd count by using the MCNN system influenced by the multi-column deep neural networks [12]. The system holds three columns of CNN with filters sizes, large, medium, and small. The arrangement of MCNN consists of Max pooling for every region.Features maps transform to thedensity map, they use filters with dimensions 1 × 1 [13]. The system calculates the final prediction by averaging the unique predictions of every deep neural network. They evaluate the design on four distinct datasets; three existing datasets and their dataset. The performances of MCNN on Shanghai tech dataset obtain values 110.2, 173.2, 26.4, 41.3 for MAE and MSE respectively.

In [14], Siyu Huang, Xi Li et al., developed the person counting method from the view of semantic modelling. Existing methods [11,15] mostly focused on modelling usingthe properties of full body or the heads, disregarding the semantic structure data. The pedestrian semantic model consists of three elements: peoples, heads, and their context structure. They formulate three elements and design them as two models: a first-named body map and context structure of body parts [16].
Existing methods like conventional density maps [17], ignored the shapes of individual pedestrians, structured density maps focus on both density distributions and shapes of humans and to model semantic structure information. They evaluate the method using four datasets, the WorldExpo'10 dataset [18], the Shanghai tech-B dataset [11], the UCSD dataset [19], and the UCF_CC_50 dataset [20].

In [21], Boominathan et al. developed a framework for calculating the density of crowd from dense crowds. Focusingon the obstacle of scale variation, the model concurrently performs at high and low-level patterns generated using deepand shallow cnn. The deep network concentrates on the desired high-level semantics required architectural design similar to the VGG-16 model [22]. Adding the total predicteddensity maps gives the total count. By using a Gaussian kernel, They also perform two types of augmentation for scalevariations in crowd images and improves CNN's performance.

In [23], Lingke Zeng et al. introduced MSCNN for static crowd counting. In prior systems [11] [21], that overcomes the problem of scale variations, but it has two weaknesses; needsa pre-trained model and requires more parameters to added computing resources. Multi-Scale Blob (MSB) is the scale feature, contains various filters with various kernel sizes. ReLU [24], which performs as the activation function of preceding convolutional layers. The ShanghaiTech and UCF CC 50 datasets. Using the ShanghaiTech dataset MCNN attains MAE and MSE values 83.8, 127.4, 17.7, and 30.2 respectively. Using UCF CC 50 dataset achieves MAE and MSE values 363.7 and 468.4.

In [25], Jianing Qiu et al. proposed a Two-Column CNN (TCCNN) to calculate highly dense crowds. The design is derived from VGG-16 and Alexnet. They joined two networks to output the density map. Before being fed into the

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

network, the system creates crowd image patches with a resolution of 224 x 224 pixels. The number of people within each patch can be obtained by finding the integral density map, and the sum of people in each patch can add up to find the total number of people. The combination of adjusted VGG-16 and Alex net gave an excellent performance. The method achieves high accuracy on the above datasets.

In [26], Youmei Zhang et al. proposed a model to focus on head positions used for crowd counting. AM-CNN contains 3 shallow layers, attention models. The system evaluates using the above 3 datasets. They use Gaussian kernels for density maps. For ShanghaiTech and UCF CC 50 datasets, the MAE values for Large, Medium, Small are 112.0, 121.1, 129.1.1
respectively, and MSE values obtained are 166.1, 200.8, 198.The performance of this model is 29.2/40.7 (L), 39.4/39.1 (M), and 32.6/29.4(S).

In [27], Hailong Li et al. proposed method Alex Net [28] with multi-layers and edge boxes. For extracting training data edge boxes were used. The INRIA pedestrian dataset, including 614 samples used for training and 218 samples for testing. Boxes algorithm achieves good results.The false rate is 10%, missing range is 23%. In [29], Joseph Redmon proposes that YOLO is swift for object detection in images.Existing methods repurpose classifiers to perform detection. Data sets from PASCAL VOC 2007 [30] and 2012. The VOC 2012 test set achieves a mAP of 57.9%.

In [32], Jiahuan Zhou et al. developed a detection system the model consists of 9 convolutional layers. The YOLO bounding boxes coordinates data contains (xcentre, ycentre,w, h)) with a confidence value [0, 1]. The pre-trainedmethod trained on VOC2007.The system obtained a MAP of 44.3% in their dataset. In [33], Igor R. de Almeida et al. developed a computer vision technique to find the dynamicchanges in crowds.

In [34], Qiming proposed a discriminative weighted sparsepartial least model for feature selection .This method is used for human detection. This applies sparse PLS forfeature selection. To create a latent matrix, formulate a discriminative regularized weighted least square problem. The training phase extracts different channel features, including LUV colour channels, gradient magnitude, and histograms to generate a large number of channel features.

The discriminative features can be selected based on the weight matrix. Evaluated the performance of the DW SPLS approach on three challenging human datasets, including INRIA [35], Caltech [36], and TUD-Brussels [37] pedestrian data sets.

In [40], Athanasios Tsitsoulis et al. presented a method for the extraction of the person in an image. For person detection, this system uses the location, dimensions, and colour of the face. The primary contribution of the method joins information from several kinds of image segmentation. First, the algorithm can assign firm edges in the image. This system combines the global detection technique with an appearance model. Evaluated the algorithm using INRIA person dataset.

In [42], Kuan-Hui Lee et al. proposed a system which detects persons from video and finds the persons. Thesystem uses the ETH Mobile Scene (ETHMS) dataset. The detection rate of the model is 75%. In [43], Shen Li et al. developed a CD-CNN, for visual tracking. The method evaluated on OTB 2015 benchmark and the OTB 2013 Benchmark. The method has 0.600 AUC value for OTB2015 and 0.627 on OTB 2013.In [44], Zheng Tang et al. proposed a video scene framework that tracks multiple human objects and estimates their 3D poses.
c). The (x, y) coordinates signify the centre of the box, comparative with the matrix cell area. The (x, y) values are normalized to [0, 1]. The (w, h) coordinates denote width andheight also normalized to [0, 1]. Here, c denotes the class of the corresponding object (1 denotes a person). The system obtains the objectness score using logistic regression.

YOLO v3 makes predictions over scales; YOLO v3 applies three various scales. The most powerful feature of YOLO v3is prediction across three distinct scales for all positions of the input image. YOLO v3 makes predictions identical to thefeature pyramid network, FPN. Every prediction is made of a boundary box, objectness, and class score. In Darknet-53 the convolutional layers are added for feature extraction. Theearlier version YOLO v2 utilized Darknet-19. YOLO is a fully convolutional network.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

III. METHODOLOGY

A. Overview

Object detection is the process of recognizing the presence, area, and class of at least one object within an image. The person detection process draws a bounding box around everyperson in the image and allots them with a 'person' label. YOLO is an object detection strategy with a deep convolutional neural network to detect the object. YOLO wasproposed by Redmon et al.[29], in the year of 2016. It looks at the full image only once then it passes through the network and detects objects.

YOLO takes an input image and divides it into S x S matrices. The centre of an object drops into a matrix region, that regionis responsible for distinguishing an object. The confidence value represents how likely the box contains an object and how exact is the bounding box. YOLO has an organization structure of 24 convolutional layers, a max-pooling layer, andtwo completely connected layers towards the end.

YOLO v2 is a version of the YOLO, proposed in the year 2017 to improve the accuracy. But, in the case of small-sized object detections YOLOv2 struggles.YOLO v3 has few incremental improvements compared to YOLO v2 [46]. YOLO v3 was introduced in April 2018 [47].

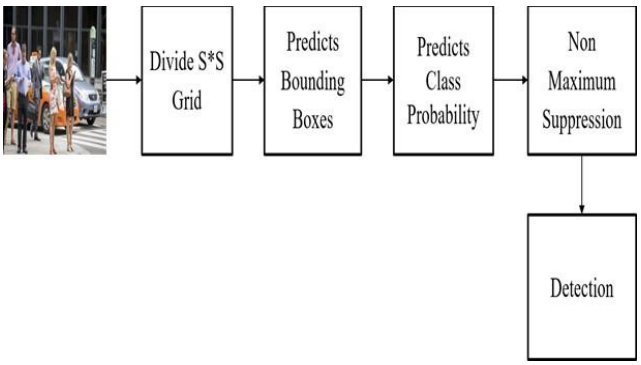

Fig.1. YOLO Person Detection flow

The proposed method begins with the dataset collection step.From the dataset images are divided into two, train dataset, and test dataset. Using 243 images from the train dataset, Yolo_mark labelling tool marks the bounding boxes for persons and obtains the corresponding annotation files. The darknet YOLO v3 framework gets trained using the weight values obtained from annotation files. After pedestrian detection, the system determines the count of people in that image. Figure.2 shows the procedure for pedestrian detectionand counting.

The proposed system is designed for pedestrian detection in images and videos using YOLO v3. YOLO v3 first takes an input image then divides it into grids, that grid region is liable for recognizing an object. YOLO v3 then figures the bounding boxes, their corresponding probabilities for personobjects.

Bounding Box Prediction is the process to draw the bounding .

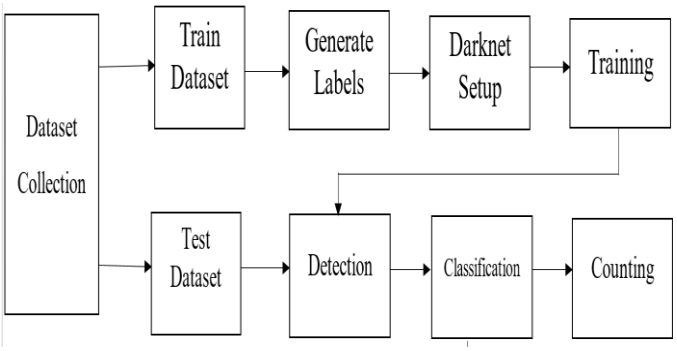

Fig.2.Represents proposed system working

boxes for each recognized object and predict confidence value, which has five components; x-coordinate, y -coordinate, width, height and class of the object (x, y, w, h, DATASET

Pedestrian detection and counting process uses an extensivecollection of images and videos. The image datasets used

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

are the INRIA dataset and Shanghai tech-B Dataset. INRIA used a static pedestrian detection data set. The INRIA dataset is categorized into two, for training, testing purposes. From the INRIA dataset, 243 positive samples which contain a total of1416 persons are used for training YOLO v3. The test set has122 positive samples which contain a total of 309 persons andinclude 58 negative samples. The negative samples do not contain pedestrians. The ground truth of the images is not available, and thus, the total count is estimated via visual inspection. Shanghai tech-B Dataset contains comments on images that were taken using various cameras in the busy streets in Shanghai. In the Shanghai tech-B dataset, for testing purposes the ground truth of the given images is given. For the testing, the system uses 56 images containing persons.

For pedestrian detection in videos, the system randomlyselected 15 videos from the internet. The system aims to detect the highest traffic experienced in different locations ina city over different periods of time. For identifying the highest density of traffic experienced in an area in a specified period, the peak count is calculated rather than the total count. The ground truth of the videos is not available, and thus the peak count is estimated via visual inspection`.

B. GENERATE LABELS

Annotation tools are used for annotating the image for training YOLO. First, each person is marked in the images, from the INRIA dataset, using visual GUI-software, and thenlabel each of them for generating annotation files. The systemuses Yolo_Mark labeling tool [48] for marking each person in that image. Yolo_Mark creates a .txt file for each .jpg image in the same folder with the same name. The .txt recordcontains the following values: <object-class> <x_center> <y_center> <width> and <height>.

The object-class denotes the class of the object using integernumbers starting from 0. This system focuses only on one class, which is a person so it is set to 0. The x_center and y_center are the x and y coordinates of the center of the bounding box, it can range from 0.0 to 1.0. Width and height denote the width, height of the bounding box also it can rangefrom 0.0 to 1.0.

Normally annotation tools follow two steps, first draw the bounding box in an image and then store top-left and bottom-right points in the corresponding text file. Then convert the points into YOLO input format. Yolo_Mark labeling tool combines these two steps. It draws the bounding boxes on image and creates a .txt-file for each .jpg image-file that contains points in the YOLO input format. Yolo_Mark calculate the annotation values using equations 3.1, 3.2, 3.3, 3.4 as follows:

Center-x = x /W (3.1)
Center-y = y/H (3.2)
Width = w/W (3.3)
Height = h/H (3.4)

Here x denotes the x-coordinate of the center of boundingbox, y denotes y-coordinate of the center of bounding box, w denotes the width of the bounding box, h denotes the height of the bounding box, W denotes the width of the whole image and H denotes the height of the whole image. For the labeling process, the system loads images from the Inria dataset to the Yolo_Mark image directory and sets the number of classes to zero and object name as a person. Then run Yolo_Mark using the command, yolo_mark.exe data/img data/train.txt data/obj.names. Then open all the images from the dataset, mark bounding boxes for each pedestrian, and finally the corresponding .txt file containing the coordinates areobtained. The .txt file contains values as follows:

0 0.508594 0.516667 0.367188 0.755556
0 0.458984 0.433333 0.280469 0.636111
0 0.499609 0.531944 0.308594 0.561111

In the first line 0 represents the value for the object class person, 0.508594 denotes the x-center value, 0.516667 denotes the y-center value, 0.367188 denotes the width and 0.755556 denotes the height.

C. TRAINING

The training process method makes use of three files .data, .names and .cfg. After creating the files of the above-mentioned extensions copy the dataset images and annotations to the corresponding directory. Using pre-trainedweights YOLO starts training [51]. Darknet [49] [50], is an open-source framework to train neural networks, which is written in C or CUDA (Compute Unified Device Architecture), and serves as the basis for implementing YOLO. It supports CPU and GPU computation and installation is simple. Darknet is easy to install as it only requires two optional dependencies, OpenCV and CUDA. OpenCV is used to increase the variety of supported image types and CUDA is used for enabling GPU computation. Clone the Darknet git repository and run make- file to compile it.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

1. YOLO v3 Configuration Parameters

Firstly, create the configuration files to train YOLO v3 for person detection. Download the yolov3.cfg file and copy the same contents to the Yolo-person.cfg file. The configuration file contains the training parameters like batch size, subdivision, etc. Here set the batch=64, the batch parameter represents the batch size. For one iteration 64 images were used to update the parameters. Then set subdivisions = 8, darknet provides a variable called subdivisions to define the part of the batch size to be used for one time on our GPU. TheGPU process batch/subdivision number of pictures at any time. The Complete iteration would be finished only after processing all the 64 images.

The parameters determine the input size and the number of channels. Assign the values for width = 416 and height = 416. If the design expands the size, it will produce good results butthe training takes more time. Then set the channels = 3, it denotes the 3 channels for processing RGB input. The momentum and decay parameters manage the weight is updated.Here momentum equal to 0.9 and decay equal to 0.0005, which controls the penalty term.

The learning parameter rate regulates aggressive methods should learn based on the current batch of data. Generally, the value ranges from 0.01 to 0.0001. In training, it starts with zero learning. The learning rate should decrease over time. The method has a below learning rate for a small period of time frame at the origin.

Data augmentation is a process to create new training data from our training data artificially. For the data augmentation process, the angle parameter in the config file will provide a random rotate to the given image by ± angle. The process transforms the colors of the whole image using saturation, exposure, and hue, it is yet an image containing the person. For training, the model system wants to determine the numberof iterations. It contains only one object class, person, so change the max_batches to (classes*2000), so set max_ batches=2000.

The training process wants to specify the number of classes used for detection. Here having only one object class, people so set classes = 1 in every YOLO layer. The process wants to determine the number of filters, set filters = (classes + 5) *3 in all the YOLO layers. So in our model value of classes=1and value of filters=18.

2. Creating Data File

For training, the process wants to create a '.data' file. Person.data file contains the number of classes, names of the object classes, t path to train and validation files, and the path to a backup file for storing the weights file. The Person.data file contains the following values:

classes equal to 1

train equal to data/train.txtvalid equal to data/test.txt

names equal to data/person. namesbackup equal to backup/

3. Creating Name File

In the data file, the line names represent the way the file consists of the name of the classes. For that, we create a file Person.names that contains the name of the objects with a newline. The named file contains only one object name, person.

4. Image and Annotation File

Copy the set of .jpg images from the INRIA dataset, used fortraining, and corresponding .txt files created by Yolo_Mark to the object directory of the darknet, build\darknet\x64\data\obj\. Then create a train.txt file in thedata directory of the darknet. The train.txt file consists of filenames of the images. Add each filename with a new line,for example;

data\person\crop_000010.pngdata\person\crop_000011.pngdata\person\crop_000012.png

5. Download Pre-trained Weight

For training purposes, the model needs the convolutional weight file. The pre-trained weights are downloaded to the darknet directory. The convolutional weights are trained on ImageNet. Finally, start training the darknet YOLO v3.The weight file will be saved in the backup file after completing the 100 iterations. The best weight used for person detection.

D. DETECTION

Person detection using YOLO-v3 requires four input arguments; Input image, YOLO-v3 configuration file, trained YOLO-v3 weights, and the text file containing class name person. Firstly, the width and the height of the input image are obtained. Then the colors for the label and the bounding boxes are applied. The function cv2.read From Darknet, creates a network using weight and configuration files. Finally, the input image passes through the deep neural network [52][53].

YOLO-v3 uses multiple output layers to obtain the prediction. The detected person region is marked by the bounding boxes. The function draw-bounding-box for draw the bounding boxes. The class label value is assigned over thebounding boxes. The confidence value obtained indicates how confident that the bounding box contains a person. Higher confidence value indicates the network trust the bounding box contains the object, lower confidence value indicates that

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

the network distrusts the objects in the bounding box. For that reason, the model considers only the objects with a confidence value greater than 0.5. Filtered outall the weak detections using the confidence value.
The same object can be detected more than one time. The model wants only one detection for one object with bounding boxes and confidence value. This is obtained using Non-maxsuppression. Non-max suppression is the process to ignore the weak detection for the same objects. There could be morethan one bounding box per person. First check the value of pc, which is the confidence of an object to be in the image, for each box then discard all boxes with pc <=0.6. Then if there are any remaining boxes left, select the box with the largest pc value. Example, if the YOLO-v3 detects the single person with three bounding boxes. The confidence value for three bounding boxes 0.6, 0.7, and 0.9 respectively. Here themodel selects the bounding box with the greatest value, 0.9 for that person.

E. GROUND TRUTH

After detecting the persons, the total count of people in that image is calculated and the model wants the actual count in the image to assess the accuracy. This actual value is named as ground truth. The ground truth of the images is not available and thus, the total count is estimated via visual inspection in the case of the INRIA dataset, video files. In the case of the Shanghai tech-B dataset, which was used for testing purposes, the ground truth of the given images is available. ShanghaiTech B-Dataset consists of 120 images and its corresponding ground-truth value as .mat files.

Filename	: GT_IMG_1.mat	Total count	: 23
Filename	: GT_IMG_10.mat	Total count	: 181
Filename	: GT_IMG_100.mat	Total count	: 157
Filename	: GT_IMG_101.mat	Total count	: 37
Filename	: GT_IMG_102.mat	Total count	: 70
Filename	: GT_IMG_103.mat	Total count	: 57
Filename	: GT_IMG_104.mat	Total count	: 44
Filename	: GT_IMG_105.mat	Total count	: 227
Filename	: GT_IMG_106.mat	Total count	: 165
Filename	: GT_IMG_107.mat	Total count	: 476
Filename	: GT_IMG_108.mat	Total count	: 139
Filename	: GT_IMG_109.mat	Total count	: 316
Filename	: GT_IMG_11.mat	Total count	: 164
Filename	: GT_IMG_110.mat	Total count	: 110
Filename	: GT_IMG_111.mat	Total count	: 20
Filename	: GT_IMG_112.mat	Total count	: 131
Filename	: GT_IMG_113.mat	Total count	: 48
Filename	: GT_IMG_114.mat	Total count	: 183
Filename	: GT_IMG_115.mat	Total count	: 101
Filename	: GT_IMG_116.mat	Total count	: 204
Filename	: GT_IMG_117.mat	Total count	: 40
Filename	: GT_IMG_118.mat	Total count	: 143
Filename	: GT_IMG_119.mat	Total count	: 146
Filename	: GT_IMG_12.mat	Total count	: 513
Filename	: GT_IMG_120.mat	Total count	: 70
Filename	: GT_IMG_121.mat	Total count	: 175
Filename	: GT_IMG_122.mat	Total count	: 51
Filename	: GT_IMG_123.mat	Total count	: 55
Filename	: GT_IMG_124.mat	Total count	: 67
Filename	: GT_IMG_125.mat	Total count	: 72
Filename	: GT_IMG_126.mat	Total count	: 56
Filename	: GT_IMG_127.mat	Total count	: 165
Filename	: GT_IMG_128.mat	Total count	: 32
Filename	: GT_IMG_129.mat	Total count	: 149
Filename	: GT_IMG_13.mat	Total count	: 48
Filename	: GT_IMG_130.mat	Total count	: 80
Filename	: GT_IMG_131.mat	Total count	: 117
Filename	: GT_IMG_132.mat	Total count	: 162
Filename	: GT_IMG_133.mat	Total count	: 137
Filename	: GT_IMG_134.mat	Total count	: 120

Fig.3.Ground Truth values for ShanghaiTech B-Dataset

IV. EXPERIMENTAL RESULT

A. REQUIREMENTS

Pedestrian detection and counting uses the software andhardware platform listed as follows;

- CMake 3.8
- CUDA 10.0
- OpenCV 3.7.1
- cuDNN 7.4 for CUDA 10.0
- MSVS 2017 (v15)
- NVIDIA GetForce 920MX

B. DATASET:

From the INRIA dataset, 243 positive samples which containa total of 1416 persons are used for training YOLO v3. The test set has 122 positive samples which contain a total of 309persons and also include 58 negative samples. The negative samples don't contain pedestrians. The negative samplescontain images of the road, desert, etc.
From Shanghaitech-B dataset 56 images containing persons used for the testing purpose, the ground truth of the given images is available. For the testing process using video is conducted over a sample of 15 videos, containing people collected randomly from the internet. The ground truth of thevideos is not available and thus the peak count is estimated via visual inspection.
Yolo_Mark is used for drawing bounding boxes of persons, gets its annotation files using 243 images from the INRIA dataset. Yolo_Mark creates a .txt-file for each .jpg image-filein the same directory with the same name. Figure 3 shows thelabeling process of the Yolo mark Labelling Tool using the Inria dataset images.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

The pedestrian detection using YOLO v3 draws a bounding box around every person in the picture, and allot with a person label, and the confidence scores each person. The detection process uses the images from the Inria dataset and ShanghaiTech B-Dataset. Detection results both Inria and ShanghaiTech B-Dataset images shown in the figures, Fig.4and Fig.5 Each person detected with the bounding boxes labeled as a person and corresponding class confidence values greater than 0.6.

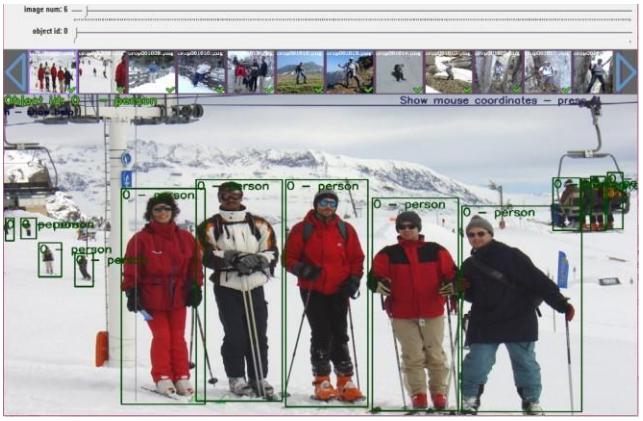

Fig.4..Generating Label Using Yolo_mark Labelling Tool

Fig.5.Using Inria dataset image persons detected with their confidence values

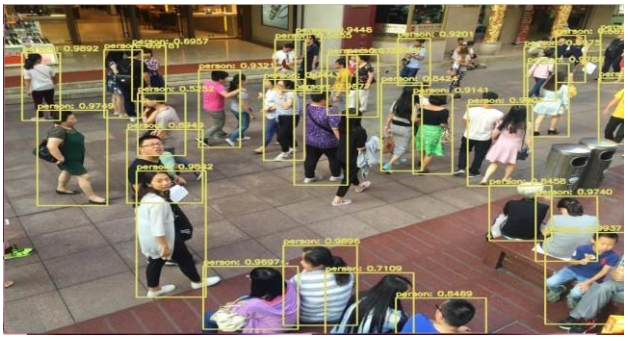

Fig.6 Using Shanghaitech-B Dataset image persons detected

After detecting the pedestrians, the total count of people in that image is calculated. In the case of videos, the peak count is calculated rather than the total count for identifying the highest density of traffic experienced in an area in a specified time period. The ground truth of the videos is not available and thus the peak count is estimated via visual inspection. Count the total number of pedestrians present in images using the Inria dataset shown in the figures Fig: 6 & Fig: 7. Calculate the total number of pedestrians present in picture using ShanghaiTech B-dataset shown in the figures, Fig: 8 &Fig: 9. Fig: 10 & Fig:11 shows the total count of pedestrians obtained using video samples.

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

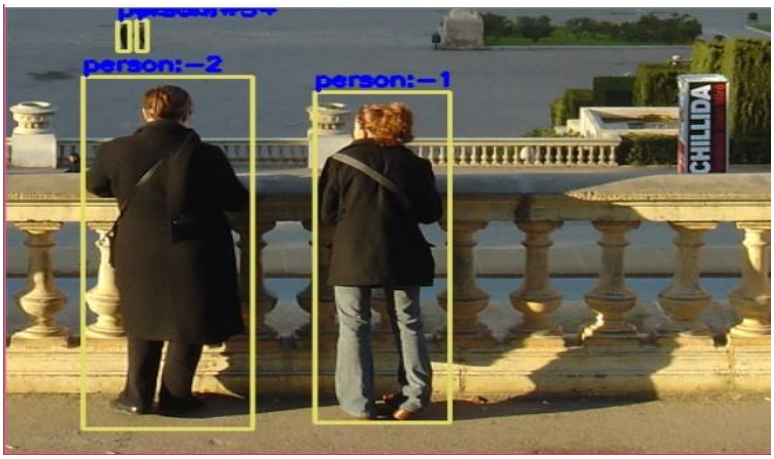

Fig: 7. Detection and counting using Inria dataset image

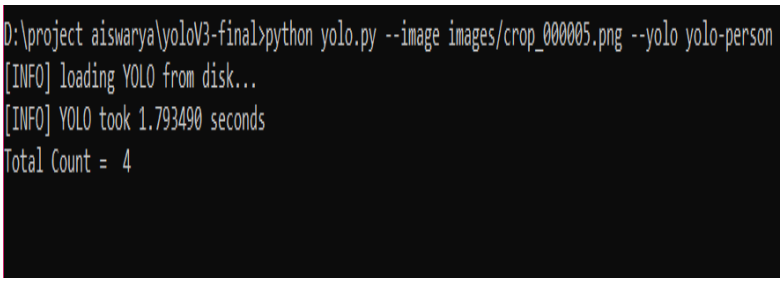

Fig:8.Total count of detected pedestrians using Inria dataset

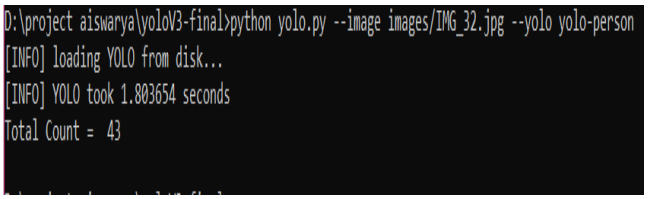

Fig:9.Total count of detected pedestrians using ShanghaiTech B-Dataset

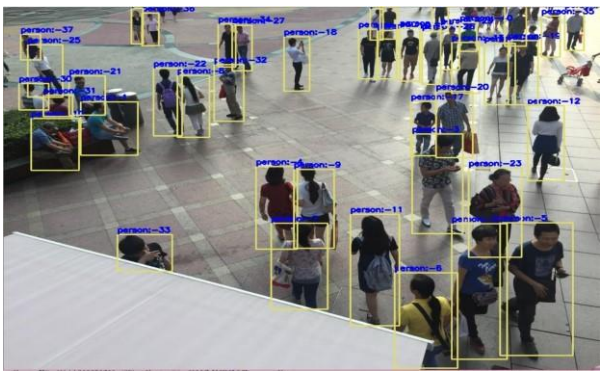

Fig.10.Detection and counting using ShanghaiTech B-Dataset image

IJARCCE ISSN (O) 2278-1021, ISSN (P) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

ISO 3297:2007 Certified ⁜ Impact Factor 7.39 ⁜ Vol. 11, Issue 5, May 2022

DOI: 10.17148/IJARCCE.2022.115172

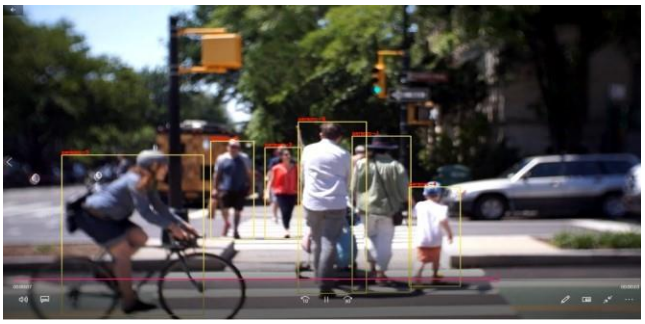

Fig.11.Detection and counting using Video samples

```
D:\project aiswarya\yoloV3-final>python yolo_video.py --input videos/family_crossing.mp4
put output/family_crossing.avi --yolo yolo-person
[INFO] loading YOLO from disk...
[INFO] 300 total frames in video
[INFO] single frame took 3.0960 seconds
[INFO] estimated total time to finish: 928.8105
Total count =  10
[INFO] cleaning up...
```

Fig:12.Total count of detected pedestrians in video

V. CONCLUSION

Object detection decreases human endeavors in numerous fields. YOLO-based Convolutional Neural Network group of models for object detection and the most recent version is called YOLOv3. Pedestrian detection and counting using YOLO v3 fastest, and one of the most accurate algorithms. The efficiencies of the yolov3 system are the greatest advantage of it, as the system only needs to go over the input only once. The results obtained in person detection using yolov3 is one of the best among the existing systems. The time taken by the system to produce the output is much less than the existing ones. Detections at 3 layers helps to address the issue of detecting very small objects compared to the previous versions of YOLO. The testing on the Inria dataset resulted in an accuracy of 96.1% whereas that on the ShanghaiTech dataset is comparatively less at 82.1% as the images are crowded.

VI. FUTURE SCOPE

The algorithm detects the persons whose full body is included in the frame, which degrade the accuracy in crowded scenes, which could be improved in the future. It can't detect duplicates when someone re-enters into the frame the person has counted again as a new person. The algorithm could be extended to overcome this problem. The training could be extended to detect persons using the head for better handling of crowded inputs. Slightest improvements in these algorithms can improve accuracy.

REFERENCES

[1] Gozde Karatas, Onder Demir and Ozgur Koray Sahingoz "Deep Learning in Intrusion Detection Systems" International Congress on Big Data, Deep Learning and Fighting Cyber Terrorism (IBIGDELFT) 2006 vol 4, Pages: 113 - 116.

[2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath et al, "Deep neural networks for acoustic modeling in speech recognition: "The shared views of four research groups, IEEE Signal Process. Mag 2012, vol. 29, no. 6, pp. 82–97.

[3] P. F. Felzenszwalb, R. B. Girshick, D. Mcallester, and D. Ramana "Object detection with discriminatively trained part-based models" IEEE Trans. Pattern Anal. Mach. Intell., 2010, vol. 32, no. 9, p. 1627.

[4] Zhong-Qiu Zhao, and Shou-tao Xu, "Object Detection with Deep Learning: A Review" IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS FOR PUBLICATION,arXiv:1807.05511v2 [cs.CV] 16 Apr 2019,.

[5] Olga Russakovsky, Jia Deng, Hao Su Jonathan Krause et al. "ImageNet Large Scale Visual Recognition Challenge" arXiv:1409.0575v3 [cs.CV] 2015.

[6] J. R. R. Uijlings, K. E. A. van de Sande T. Gevers, A. W. M. Smeulders "Selective Search for Object Recognition" International Journal of Computer Vision September 2013 Volume 104, Issue 2, pp 154–171.

[7] Ross Girshick Jeff Donahue Trevor Darrell Jitendra Malik UC Berkeley "Rich Feature Hierarchies For Accurate

DOI: 10.17148/IJARCCE.2022.115172

Object Detection And Semantic Segmentation", arXiv:1311.2524v5 [cs.CV] 22 Oct 2014.

[8] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks", 2012 in NIPS.

[9] Ross Girshick ,Microsoft Research"Fast R-CNN" arXiv:1504.08083v2 [cs.CV] 27 Sep 2015.

[10] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks" arXiv:1506.01497v3 [cs.CV] 6 Jan 2016.

[11] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network"in Proc. IEEE Conf. CVPR, June 2016, pp. 589–597.

[12] D. Ciresan, U. Meier, and J. Schmidhuber "Multi-column deep neural networks for image classification" IEEE 2012, In CVPR, pages 3642–3649.

[13] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation", arXiv:1411.4038, 2014.

[14] Siyu Huang, Xi Li, Zhongfei Zhang, Fei Wu, Shenghua Gao, Rongrong Ji, and Junwei Han "Body Structure Aware Deep Crowd Counting" IEEE Transactions On Image Processing, Vol.27, No. 3, March 2018.

[15] C. Zhang, H. Li, X. Wang, and X. Yang "Cross-scene crowd counting via deep convolutional neural networks" in Proc. IEEE Conf. CVPR, June 2015, pp. 833–841.

[16] P. Luo, X. Wang, and X. Tang "Pedestrian parsing via deep decompositional network" in Proc. IEEE ICCV, Dec. 2013, pp. 2648–2655.

[17] V. Lempitsky and A. Zisserman "Learning to count objects in images", in Proc. Adv. NIPS, pp. 1324–1332, 2010.

[18] C. Zhang, H. Li, X. Wang, and X. Yang," Cross-scene crowd counting via deep convolutional neural networks" in Proc. IEEE Conf. CVPR, June 2015, pp. 833–841.

[19] H. Idrees, I. Saleemi, C. Seibert, and M. Shah "Multi-source multi-scale counting in extremely dense crowd images" in Proc. IEEE Conf. CVPR, June 2013, pp. 2547–2554.

[20] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos "Privacy- preserving crowd monitoring: Counting people without people models or tracking" in Proc. IEEE Conf. CVPR, Jun 2008. pp. 1–7.

[21] L. Boominathan, S. S. Kruthiventi, and R. V. Babu "Crowdnet: a deep convolutional network for dense crowd counting" in Proceedings of the 2016 ACM on Multimedia Conference, pp. 640–644, ACM.K. Simonyan and A. Zisserman" Very deep convolutional networks for large-scale image recognition" arXiv:1409.1556,.3.1.1 2014.

[22] Lingke Zeng, Xiangmin Xu, Bolun Cai, Suo Qiu, Tong Zhang "Multi-Scale Convolutional Neural Networks For Crowd Counting" arXiv:1702.02359v1 [cs.CV] 8 Feb 2017.

[23] Vinod Nair and Geoffrey E Hinton (2010), "Rectified linear units improve restricted Boltzmann machines," in Proceedings of the 27th international conference on machine learning (ICML-10), pp. 807–814.

[24] Jianing Qiu, Wanggen Wan, Haiyan Yao, Kang Han "Crowd counting and density estimation via two-column convolutional neural network" 4th International Conference on Smart and Sustainable City ICSSC 2017.

[25] Youmei Zhang, Chunlin Zhou, Faliang Chang, and Alex C. Kot,"Attention to Head Locations for Crowd Counting" arXiv:1806.10287v1 [cs.CV] 27 June 2018.

[26] Hailong Li, Zhendong Wu, Jianwu Zhang "Pedestrian Detection Based on Deep Learning Model" 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics(CISP-BMEI 2016).

[27] A. Krizhevsky, I. Sutskever, G E. Hinton "Imagenet classification with deep convolutional neural networks", Advances in neural information processing systems,2012 1097- 1105.

[28] Joseph Redmon, Santosh Divvala , Ross Girshick, Ali Farhadi "You Only Look Once: Unified, Real-Time Object Detection" arXiv:1506.02640v5 [cs.CV] May 2016.

[29] M. Everingham, S. M. A. Eslami, L. Van Gool, C. K. I. Williams,
J. Winn, and A. Zisserman, "The pascal visual object classes challenge: A retrospective" International Journal of Computer Vision, 111(1):98–136, Jan 2015.

[30] Wenbo Lan, Jianwu Dang, Yangping Wang and Song Wang "Pedestrian Detection Based on YOLO Network Model" Proceedings of IEEE International Conference on Mechatronics and Automation August 2018 5 - 8, Changchun, China.

[31] Jiahuan Zhou, Lihang Feng, Ryad Chellali, and Haonan Zhu, "Detecting and tracking objects in HRI: YOLO networks for the NAO I See You function" 27th IEEE International Symposium on Robot and Human Interactive Communication, Nanjing, China, August 27-31 2018.

[32] Igor R. de Almeida, Vinicius J. Cassol, Norman I. Badler, SoraiaR. Musse, Claudio R. Jung "Detection of Global and Local Motion Changes in Human Crowd" IEEE Transactions On Circuits And Systems For Video Technology,2016.

[33] Qiming Li, Yan Yan, Hanzi Wang "Discriminative Weighted Sparse Partial Least Squares for Human Detection" IEEE Transactions On Intelligent Transportation Systems, Vol. 17, No.4, April 2016.

[34] N. Dalal and B. Triggs "Histograms of oriented gradients for human detection" in Proc. IEEE Conf. Comput. Vis. Pattern Recog., San Diego, CA, USA, 2005, pp. 886–893.

DOI: 10.17148/IJARCCE.2022.115172

[35] P. Dollr, C. Wojek, B. Schiele, and P. Perona "Pedestrian detection: An evaluation of the state of the art" IEEE Trans. Pattern Anal. Mach. Intell, vol. 34, no. 4, pp. 743–761, Apr 2002.
[36] C. Wojek, S. Walk, and B. Schiele "Multi-cue onboard pedestrian detection" in Proc. IEEE Conf. Comput. Vis. Pattern Recog, Miami Beach, FL, USA,2009, pp. 794–801.
[37] Hyeok-June Jeong, Kyeong-Sik Park, Young-Guk Ha "Image Preprocessing for Efficient Training of YOLO Deep Learning Networks" IEEE International Conference on Big Data and SmartComputing,2018.
[38] Zihan Ni1, Jia Chen1, Nong Sang1, Changxin Gao1, Leyuan Liu2"Light Yolo For High-speed Gesture Recognition" 25th IEEE International Conference on Image Processing (ICIP), 2018.
[39] Athanasios Tsitsoulis, Nikolaos G. Bourbakis "A Methodology for Extracting Standing Human Bodies From Single Images, IEEE Transactions On Human-Machine Systems, Vol. 45, No. 3,June 2015.
[40] Pengpeng Liang, Yu Pang, Chunyuan Liao, Xue Mei, and Haibin Ling "Adaptive Objectness for Object Tracking" IEEE Signal Processing Letters, Vol. 23, No. 7, July 2016.
[41] Kuan-Hui Lee, Jenq-Neng Hwang, Fellow "Ground-Moving- Platform-Based Human Tracking Using Visual SLAM and Constrained Multiple Kernels" IEEE Transactions On Intelligent Transportation Systems, Vol. 17, No. 12, December 2016.
[42] Shen Li, Bingpeng Ma,Hong Chang Shiguang Shan, Xilin Chen "Continuity-discrimination Convolutional Neural Network For Visual Object Tracking" IEEE International Conference on Multimedia and Expo (ICME), 2018.
[43] Zheng Tang, Renshu Gu, Jenq-Neng Hwang "Joint Multi-view People Tracking And Pose Estimation For 3d Scene Reconstruction" IEEE International Conference on Multimedia and Expo (ICME), 2018.
[44] Jing Tao, Hongbo Wang, Xinyu Zhang, Xiaoyu Li, Huawei Yang "An object Detection System Based on YOLO in Traffic Scene"6th International Conference On Computer Science And Network Technology (ICCSNT), 2017.
[45] Joseph Redmony, Ali Farhadiy, University of Washington and Allen Institute for AIy "YOLO9000: Better, Faster, Stronger" arXiv:1612.08242v1 [cs.CV] 25 Dec 2016.
[46] Joseph Redmon, Ali Farhadi, University of Washington "YOLO v3: An Incremental Improvement" arXiv:1804.02767v1 [cs.CV]8 Apr 2018
[47] For generating labels using Yolo mark available on https://github.com/AlexeyAB/Yolo_mark
[48] Darknet framework available on https://pjreddie.com/darknet/yolo/
[49] Darknet framework available on https://pjreddie.com/darknet/
[50] For training YOLO v3 under darknet framework available on https://github.com/AlexeyAB/darknet#how-to-train-to-detect- your-custom-objects.
[51] For pedestrian detection using YOLO v3 available on https://www.arunponnusamy.com/yolo-object-detection-opencv-python.html.
[52] For object detection using YOLO v3 available on https://www.pyimagesearch.com/2018/11/12/yolo-object- detection-with-opencv/.