Human Detection and Counter System

Dr. R. Manikandan[1], Sahaj Rohilla[2], Shubhashree Naskar[3]

[1]Assistant Professor, Department of CSE, SRM Institute of Science and Technology, Ghaziabad
[2]Department of CSE, SRM Institute of Science and Technology, Modinagar, Ghaziabad
[3]Department of CSE, SRM Institute of Science and Technology, Modinagar, Ghaziabad

---***---

ABSTRACT: One of the biggest challenges in the retail industry is measuring how many customers enter a store throughout the day. In this pandemic situation too, maintaining track of people movements and social distancing rules in a commercial setting is crucial in the current circumstances. It is a system where we use a webcam or we can give our video or images to count the number of people in a frame and if the number of people in the frame exceeds a certain limit, then it gives an alert. In this project, we are using YOLO v3 for human detection. It is an object detection algorithm which is pre-trained on the COCO image dataset. The detections are very fast and thus can be used in a real-time scenario.

Keywords: YOLO v3, COCO, Darknet, OpenCV, CNN, Human detection, Counter System

1. INTRODUCTION

Object detection is a computer vision technique that gives us various features that helps us in identifying the objects and locating them in an image or a video. With this method of identification, detecting object can be used to track the objects in a scene and determine their accurate location in a frame. The applications of object detection span multiple and diverse industries, from round-the-clock surveillance to real-time vehicle detection in smart cities. Human Detection uses computer vision to detect humans within a video. The detection of object in real-time scenarios is significant trend in industries from various cities for the surveillance.
This can be used in:

- Counts the pedestrians along a path.
- Analyzing shopper behavior or dwell time.
- Home or shop security cameras detecting intruders or visitors, etc.

Once object is detected, that moving or still object could be identified as a human being or any other thing using texture-based, shape-based or motion-based features.

The solutions for human detection have restrictions. For instance, people must be moving so that the system can distinguish other things and people, the object background must be plane or simple, or the resolution of image must be high. However, real-time scenarios always have both stationary and moving object, the object background might be complicated, and almost 80% videos in a visual surveillance system have a relatively low resolution.

In this pandemic, one of the biggest challenges in the retail industry is measuring how many customers enter a store throughout the day. Maintaining track of people movements and social distancing rules in a commercial setting is crucial in the current pandemic situation. People counting system not only just serve as an emergency plan for any pandemics, it also have range of benefits for different businesses in the long-term like, libraries, schools, airports, malls, etc. This system helps to track how many people are there in a frame. If the number exceeds a certain number, it raises an alarm.

2. LITERATURE SURVEY

Object Detection: There are large data related to object detection, some of these can be found in [1], [2], [3]. Some of the popular object detection methods uses Selective Search [4], sliding windows in Edge Box [6]. and CPMC [5]. Detecting an object mainly contains two things, the first is to locate the object and the second is to classify the object in a different class. Previous algorithms are mainly focused on face detection. Afterward, more challenging and realistic face detection datasets were created.

Histogram of Oriented Gradients method (HOG) [9]: It is one of the well-known human detection methods. It is a feature descriptor made of small regions of gradient orientation called. This method performs well if there is large number of images for training is giver, therefore needs a very careful selection of different training images [10].

Human Detection using a combination of face, head, and shoulder detection: Human detection can be done by identifying different body parts separately and combining them into one detection as in [15]. These different methods such as Haar classifier, gradient maps, golden ratio, etc., are used for detecting face, head, and shoulder. The detection results were efficient but the time of detection was high.

R-CNN: Selecting large number of regions is very difficult problem. To solve this problem, R-CNN was introduced. In this method, the author uses selective search approach, in which first it extracts just 2000 regions from the frame, called the region proposals [11]. But this method took approximately 47 seconds to give the detection and needed more amount of training time than other algorithms.

Fast R-CNN: Same author Ross Girshick, solved the drawbacks of Region Based Convolutional Neural Networks(R-CNN) algorithm to build a algorithm that is more faster for object detection and it was called Fast R-CNN [12]. Now, in this, we fed the input video or image to the convolutional Neural Networks to generate a convolutional feature map. From there they identified the region proposals [11] and wrap them into boxes. But the results were not satisfying.

Faster R-CNN: Both RCNN and Fast R-CNN algorithms uses selective search approach, thus they were slow when real-time object detection was the concern. Therefore, yet another algorithm was proposed, called the Faster R-CNN [14], in which the selective search algorithm was replaced by the object detection algorithm and it let the network learn the region proposals.

3. PROPOSED ARCHITECTURE

In this section, we discuss the methodology of our proposed work. The block diagram of the Human Detection and Counter System is shown below.

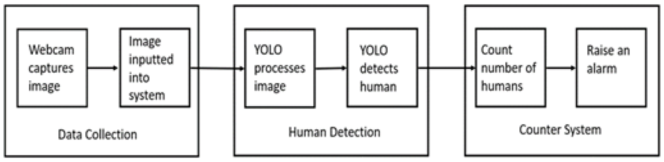


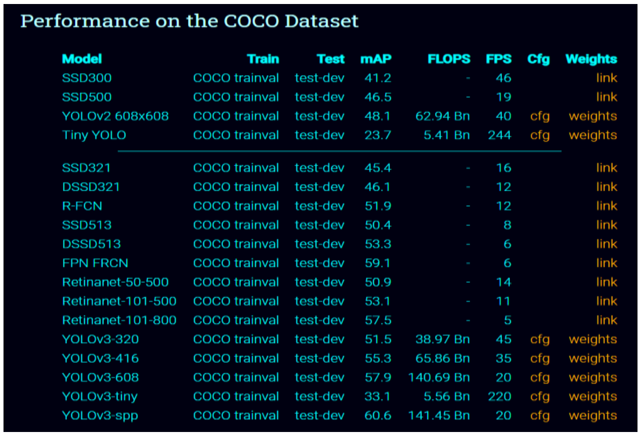
Fig. 1: Architecture of the system

When the video is captured in the webcam, the frozen frame of the video is fed into the YOLO v3 system for processing. After the image is inputted, the YOLO v3 calculates the distributed weight of the image and identifies any instance of a human. Once the humans in the frame are detected, then in the third phase it will count the number of humans in that frame. If the number of humans reaches a certain limit the systems raise an alarm or notify the user.

3.1. DATASET

We are using COCO (Common Objects in Context) [16] dataset in our project. This dataset consists of everyday objects captured from everyday scenes in their natural context. The COCO dataset consists of 80 labels, including, people, automobiles, animals, kitchen objects, dining objects, etc.

It is a very large-scale dataset that is used for object detection which consists of 330K images, 250,000 people with key points and 1.5 million object instances.



Performance on the COCO Dataset

Model	Train	Test	mAP	FLOPS	FPS	Cfg	Weights
SSD300	COCO trainval	test-dev	41.2	-	46		link
SSD500	COCO trainval	test-dev	46.5	-	19		link
YOLOv2 608x608	COCO trainval	test-dev	48.1	62.94 Bn	40	cfg	weights
Tiny YOLO	COCO trainval	test-dev	23.7	5.41 Bn	244	cfg	weights
SSD321	COCO trainval	test-dev	45.4	-	16		link
DSSD321	COCO trainval	test-dev	46.1	-	12		link
R-FCN	COCO trainval	test-dev	51.9	-	12		link
SSD513	COCO trainval	test-dev	50.4	-	8		link
DSSD513	COCO trainval	test-dev	53.3	-	6		link
FPN FRCN	COCO trainval	test-dev	59.1	-	6		link
Retinanet-50-500	COCO trainval	test-dev	50.9	-	14		link
Retinanet-101-500	COCO trainval	test-dev	53.1	-	11		link
Retinanet-101-800	COCO trainval	test-dev	57.5	-	5		link
YOLOv3-320	COCO trainval	test-dev	51.5	38.97 Bn	45	cfg	weights
YOLOv3-416	COCO trainval	test-dev	55.3	65.86 Bn	35	cfg	weights
YOLOv3-608	COCO trainval	test-dev	57.9	140.69 Bn	20	cfg	weights
YOLOv3-tiny	COCO trainval	test-dev	33.1	5.56 Bn	220	cfg	weights
YOLOv3-spp	COCO trainval	test-dev	60.6	141.45 Bn	20	cfg	weights

Fig. 2: Performance on the COCO Dataset [16]

3.2. YOLO v3

YOLO, it is an algorithm used for object detection which stands for You Only Look Once. It is extremely fast and accurate. We can easily customize our preference between accuracy or speed by changing the size or the frames of the model. It applies single neural network to the full image. Then the image is divided into small regions by network and predicts bounding boxes and probabilities for every region. After taking the input video or image, the YOLOv3 algorithm will calculate the distributed weight of the image to locate the individual objects in the frame. When the system detects human, then the system counts the number of the human in the frame. These bounding boxes that are filtered by network layer have some predicted probabilities. YOLO splits the image up into regions, and divides frame into S*S grids of cells, and subsequently finds the bounding boxes and confidence within the input image. The bounding boxes are weighted by the probabilities and the model makes its detection based on the final weight.

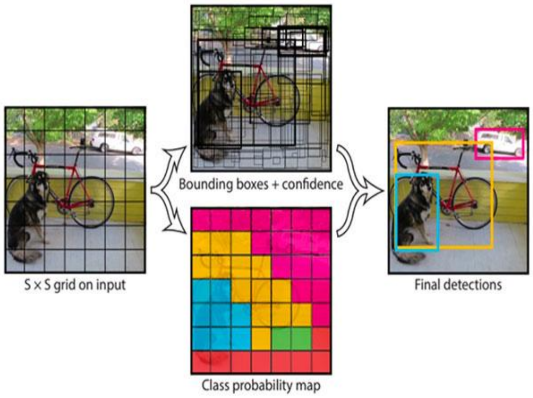


Fig. 3: YOLO model **[17]**

3.3. DARKNET

Darknet is neural network which is written in C and CUDA. It is an open-source. YOLO uses weights of the darknet53 model. For detection, 53 more layers are stacked on previous darknet53 layers, so it is 106-layer architecture for YOLOv3.That makes YOLO a large architecture and enhancing accuracy at the same time. In YOLOv3 detections are done at three different layers. On 82nd layer the first detection is made then, on 94th layer the second detection is done. The last detection is made by the 106th layer.

4. SETUP

We loaded YOLOv3 and Dataset (COCO) [14] through the framework OpenCV.

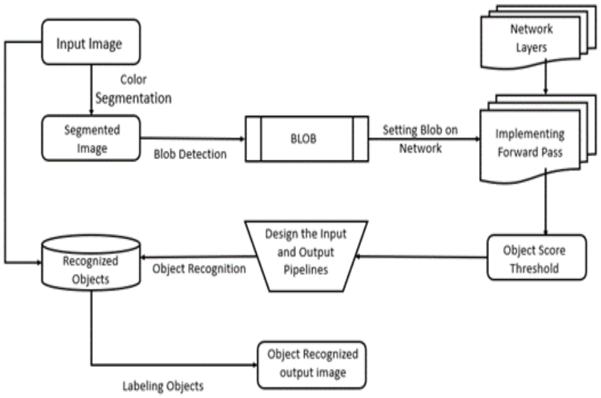


Fig. 4: Flow diagram of YOLO

The above figure shows the process of detecting an object in yolo. First an input image is fed into the system. Color segmentation occurs in that image. This segmented image passes through the BLOB detector. In the network layers the Darknet is already added and then these layers are implemented on the image which came out of the BLOB detector. After that an object score is given to each object. Input and Output is designed and finally the object is detected.

The Human Detection and Counter System works in different phases. First, we get detection of all the objects present in the frame. Then we assign random colors to every class. Like, there are 80 classes so 80 random color are assigned. Then we have to convert the image into the blob. Blob extracts the features from the image. (416,416) is the standard size, then this blob image is pass through the algorithm. If the confidence is more than 50% then the object is detected. Then a rectangle is drawn around the object.

If the object detected is a human, then it is labelled as 'person'. Once the humans are detected the distance between them is calculated using Euclidean Distance.

$$d(\mathbf{p},\mathbf{q}) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2}$$

p, q = two points in Euclidean n-space

q_i, p_i = Euclidean vectors, starting from the origin of the space (initial point)

n = n-space

Fig. 5: Distance formula

If two people are too close then the rectangle changes the color to red or yellow and if the person is far away from one another then it changes to green. A line is drawn between them which calculates the distance between them. Along with this there is a counter system, which gives the information about how many people are there in the frame at a time. Finally, there is an alert system, which informs the user whenever the number of people exceed a certain limit in the area.

5. RESULT AND CONCLUSION

In this paper, we are using deep learning and image processing algorithm, namely You Only Look Once (YOLO) algorithm, for the purpose of human detection in real-time. The detections use COCO dataset. We performed the detections on images, videos and using live camera as well. The system is capable of detecting humans. The detections are fast and can be used in real-time. The alert system alerts whenever the number of people exceed 5. We can change this limit. The algorithm is simple to build and can be trained directly on a complete image. The Human detection and counter system can successfully detect human beings in real-time scenarios areas like complicated scenes as well as effectively identify them even in circumstances of occlusions. The system also checks, whether the people in the frame are following social distancing norms or not. The system also has an alert system. It raises an alarm when the number of people exceed a limit. This system provides us with an inexpensive human counter.

6. REFERENCES

[1] J. Hosang, R. Benenson, and B. Schiele, "How good are detection proposals, really?" in British Machine Vision Conference (BMVC), 2014.

[2] J. Hosang, R. Benenson, and B. Schiele, "How good are detection proposals, really?" in British Machine Vision Conference (BMVC), 2014.

[3] N. Chavali, H. Agrawal, A. Mahendru, and D. Batra, "Object-Proposal Evaluation Protocol is 'Gameable'," arXiv: 1505.05836, 2015.

[4] J. R. Uijlings, K. E. van de Sande, T. Gevers, and A. W. Smeulders, "Selective search for object recognition," International Journal of Computer Vision (IJCV), 2013.

[5] J. Carreira and C. Sminchisescu, "CPMC: Automatic object segmentation using constrained parametric min-cuts," IEEE

Transactions on Pattern Analysis and Machine Intelligence (TPAMI), 2012.

[6] C. L. Zitnick and P. Dollar, “Edge boxes: Locating object ´ proposals from edges,” in European Conference on Computer Vision (ECCV), 2014.

[7] E. Hjelmas and B. Low, “Face detection: A survey,” ˚ CVIU, vol. 83, no. 3, pp. 236–274, 2001.

[8] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labelled faces in the wild,” University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.

[9] Ngo-Doanh Nguyen, Duy-Hieu Bui, Xuan-Tu Tran, "A Novel Hardware Architecture for Human Detection using HOG-SVM Co-Optimization", 2019 IEEE Asia Pacific Conference on Circuits and Systems (APCCAS).

[10] N. Dalal, “Finding People in Images and Videos,” PhD Thesis, Grenoble Institute of Technology, July 2006.

[11] Ross Girshick, Jeff Donahue, Trevor Darrell, Jitendra Malik, "Rich Feature Hierarchies for Accurate Object Detection and Semantic Segmentation" 2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

[12] Ross Girshick, "Fast R-CNN", 2015 IEEE International Conference on Computer Vision (ICCV).

[13] Shaoqing Ren, Kaiming He, Ross Girshick, Jian Sun, "Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks", 2017 IEEE Transactions on Pattern Analysis and Machine Intelligence.

[14] Enjia Chen, Xianghong Tang, Bowen Fu, "A Modified Pedestrian Retrieval Method Based on Faster R-CNN with Integration of Pedestrian Detection and Re-Identification", 2018 International Conference on Audio, Language and Image Processing (ICALIP).

[15] Feng Su, Gu Fang, Ju Jia Zou, "Human detection using a combination of face, head and shoulder detectors", 2016 IEEE Region 10 International Conference TENCON.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollar, and C Lawrence ´ Zitnick. Microsoft coco: Common objects in context. In European conference on computer vision, pages 740–755. Springer, 2014.

[17] Joseph Redmon, Santosh Divvala, Ross Girshick, Ali Farhadi, “You Only Look Once: Unified, Real-Time Object Detection”.