A Novel YOLO-based Real-time People Counting Approach

Peiming Ren[∓], Wei Fang[∓]* and Soufiene Djahel[±]
[∓]Department of Computer Science and Technology, School of IoT Engineering, Jiangnan University, Wuxi, Jiangsu, China
[±]School of Computing, Mathematics and Digital Technologies, Manchester Metropolitan University, UK
{peimingren@163.com, fangwei@jiangnan.edu.cn, s.djahel@mmu.ac.uk}

***Abstract*—Real-time people counting from video records is a main building bloc for many applications in smart cities. In practice, this task usually encounters many problems, like the lack of real-time processing of the recorded videos or the occurrence of errors due to irrelevant people being counted. To overcome the above issues, we propose a novel real-time people counting approach dubbed YOLO-PC (YOLO based People Counting).**

***Index Terms*—People-counting; Boundary-selection; YOLO**

I. YOLO-PC: AN OVERVIEW

In this work we modify the pioneer object detection system YOLO [1] [2] by proposing the so-called YOLO-PC (YOLO based People Counting). YOLO-PC extends the original YOLO system using a deep learning approach to achieve more accurate people counting. Due to its low computation overhead, compared to other state-of-the-art object detection systems, such as R-CNN [3], Fast R-CNN [4], and Faster R-CNN [5], and its ability to detect objects in real-time, YOLO has been chosen as the base approach in our YOLO-PC.

In terms of real-time performance, YOLO-PC re-trains a deep convolution neural network to detect people at more than 40 fps (frames per second) with the support of a GPU. Regarding people counting method, on one hand, YOLO-PC divides the image into a 9*9 grid and makes use of the boundary, which leads to more detected boxes and higher average confidence value. On the other hand, YOLO-PC chooses different boundary areas according to the actual application scenario to count people contrapuntally and further improves the counting accuracy. YOLO-PC can ignore the invalid persons who may be in the billboards or in other irrelevant areas.

Experimental results show that YOLO-PC can quickly count people with high accuracy at the entrance or exit of some places, such as escalators, scenic spots etc.

II. YOLO-PC: THE MAIN STEPS

The operation of YOLO-PC consists in five steps as shown in Figure 1. The first step consists in setting the detection threshold and adjusting the camera. The detection results below the threshold, which is usually set from 0.2 to 0.4, will not be counted. For the sake of simplicity, we use the default value of 0.2 in this work. In the actual scene, the camera should be adjusted to the appropriate height and angle.

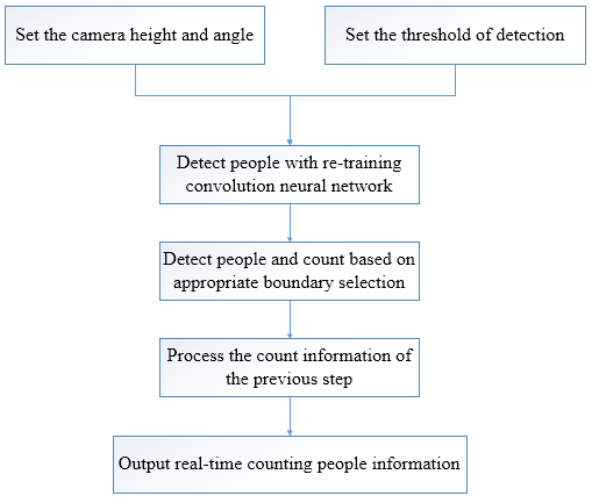


Fig. 1: The process of YOLO-PC

In the second step, we detect people through re-training a convolutional neural network. YOLO divides the image into a 7*7 grid and for each grid cell predicts two bounding boxes as well as the confidence value for those boxes. We believe that this division is not sufficient and we aim that our algorithm will be more efficient in identifying people to achieve higher counting accuracy. In other words, YOLO-PC works better with more detected boxes and higher confidence values. To this end, YOLO-PC uses 9*9 grid and 3 bounding boxes. We set up three sets of experiments of 4 minutes video each using different thresholds (i.e., 0.2, 0.3 and 0.4). The obtained results, shown in Table 1, are promising since YOLO-PC detects more boxes and achieves higher confidence values for those boxes compared to YOLO. More specifically, YOLO-PC detects more than 10 percent of the boxes, the confidence average value is more than 50 percent higher when the threshold is 0.2.

The third step consists in detecting and counting people based on an appropriate boundary selection. YOLO-PC selects one or more grid cells as the area boundary from 243(9*9*3) cells and chooses a different boundary according to the actual situation. If people turn left by somewhere, the boundary of the left area of the video should be selected, the value of the

TABLE I: The number of detected boxes and the average confidence value at different threshold values (T)

Method	Boxes number			Average confidence value		
	T = 0.2	T = 0.3	T = 0.4	T = 0.2	T = 0.3	T = 0.4
YOLO	13400	9423	5856	0.39	0.45	0.51
YOLO-PC	14664	13612	12514	0.59	0.62	0.64

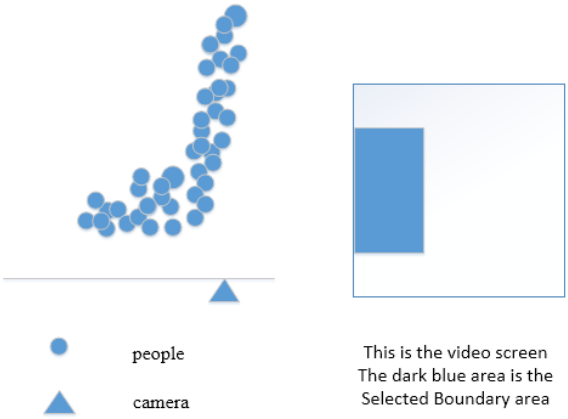


Fig. 2: Left boundary selection sketch map

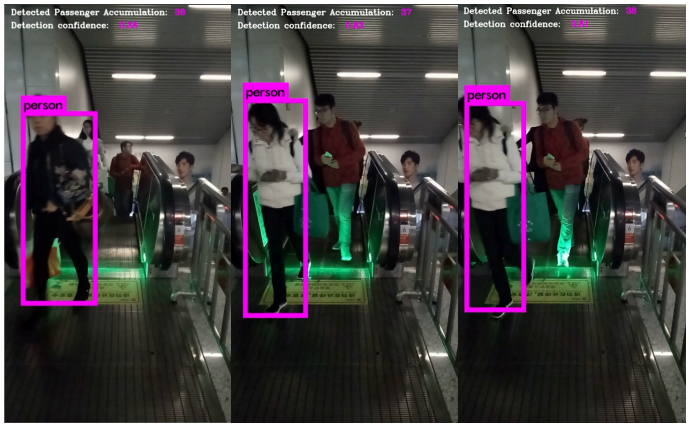
boundary is around 113 and people through that boundary will be counted. Figure 2 shows the sketch map of the left boundary area and Figure 3 shows the experiment result. Similarly, if people turn right by somewhere, the value of the boundary is around 129. If people go straight by somewhere, the value of the boundary is around 121. YOLO-PC can be more accurate in detecting the flow of people as some nonsense interference, such as people in the billboards and unrelated backs, can be ignored because of the boundary selection.

The counting information of the previous step is now processed in the fourth step. In the selected boundary area, the boxes number accumulates and constantly updates, we refer to this number as S. The value of S at a moment t in a video represents the number of detected people at that moment, which is very accurate. The value of S in a period of time represents the total number of detected people. Because it takes time for people to move in the boundary area, the value of S is repeated, that is to say, the same person has been detected many times. According to the experiments, every person has been detected around 18 times when passing through the selected boundary area, so the predicted number is $S/18$ at the default threshold.

In the fifth and last step, we output the real-time people counting information. YOLO-PC can directly show the real-time people counting information in the video images, including the current number, FPS, confidence value etc. YOLO-PC can also save real-time information and continue to update, and then output them through some interfaces.

III. Conclusion and Future Work

In this paper we introduced YOLO-PC, a YOLO based real-time people counting approach using boundary selection.



(a) The 36th person (b) The 37th person (c) The 38th person

Fig. 3: The experimental results of the left boundary selection: the experimental video is a continuous high-definition video at 30 fps, YOLO-PC continuously detects people and counts them exactly at 40 fps

YOLO-PC outperforms YOLO as it re-trains YOLO network, which enables it to detect more boxes and achieve higher average confidence value. The boundary selection in YOLO-PC makes the counting more targeted and its result accurate and fast. In conclusion, this method is very effective and it is also able to recognize irrelevant people and ignore them in the counting process. YOLO-PC has a wide range of applications as it can assist the construction of many aspects of the smart cities. On the current basis, we look forward to making more improvements to this method by adding, for example, abnormal behavior detection and children counting.

Acknowledgment

This work was partially supported by the National Natural Science foundation of China (Grant Nos. 61673194, 61105128Key Research and Development Program of Jiangsu Province, China (Grant No. BE2017630), the Postdoctoral Science Foundation of China (Grant No. 2014M560390), Six Talent Peaks Project of Jiangsu Province (Grant No. DZXX-025), Natural Science Foundation for College and Universities in Jiangsu Province (Project Number: 16KJB520051).

References

[1] J. Redmon et al., "You Only Look Once: Unified, Real-Time Object Detection," [EB/OL].https://pjreddie.com/darknet/yolov1/.

[2] J. Redmon, et al., "You only look once: Unified, real-time object detection," [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016: 779-788.

[3] R. Girshick, et al., "Rich feature hierarchies for accurate object detection and semantic segmentation," [C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2014: 580-587.

[4] R. Girshick, "Fast r-cnn," [C]//Proceedings of the IEEE international conference on computer vision. 2015: 1440-1448. fig

[5] S. Ren et al., "Faster R-CNN: Towards real-time object detection with region proposal networks," [C]//Advances in neural information processing systems. 2015: 91-99.0