mathematics

Article

A Soft-YoloV4 for High-Performance Head Detection and Counting

Zhen Zhang [1], Shihao Xia [1], Yuxing Cai [1], Cuimei Yang [1] and Shaoning Zeng [2,*]

[1] School of Computer Science and Engineering, Huizhou University, Huizhou 516007, China; zzsjbme@sjtu.edu.cn (Z.Z.); mr123zhang@gmail.com (S.X.); hzucyx@gmail.com (Y.C.); meikoyoung1024@gmail.com (C.Y.)

[2] Yangtze Delta Region Institute, University of Electronic Science and Technology of China, Huzhou 313000, China

* Correspondence: zeng@csj.uestc.edu.cn; Tel.: +86-177-5723-7213

Abstract: Blockage of pedestrians will cause inaccurate people counting, and people's heads are easily blocked by each other in crowded occasions. To reduce missed detections as much as possible and improve the capability of the detection model, this paper proposes a new people counting method, named Soft-YoloV4, by attenuating the score of adjacent detection frames to prevent the occurrence of missed detection. The proposed Soft-YoloV4 improves the accuracy of people counting and reduces the incorrect elimination of the detection frames when heads are blocked by each other. Compared with the state-of-the-art YoloV4, the AP value of the proposed head detection method is increased from 88.52 to 90.54%. The Soft-YoloV4 model has much higher robustness and a lower missed detection rate for head detection, and therefore it dramatically improves the accuracy of people counting.

Keywords: head detection; YoloV4; NMS; soft-NMS; people counting

check for updates

Citation: Zhang, Z.; Xia, S.; Cai, Y.; Yang, C.; Zeng, S. A Soft-YoloV4 for High-Performance Head Detection and Counting. *Mathematics* **2021**, *9*, 3096. https://doi.org/10.3390/math9233096

Academic Editor: Radu Tudor Ionescu

Received: 26 October 2021
Accepted: 28 November 2021
Published: 30 November 2021

1. Introduction

People counting is a process of counting the number of people in images. It is one of the most important features in a modern intelligent camera. Without this artificial intelligence technique, we have to manually count the number of people in the surveillance video. However, this is unacceptable due the fact that the scale of video data becomes larger and larger. What is worse, it is unlikely to have a precise count when the number of people is too large. For this reason, many automatic people counting methods have been proposed based on the detection of skin color [1], facial features [2], and pedestrians [3]. Nowadays, deep learning, image recognition, and other artificial intelligence (AI) technologies are continuously developing [4]. These intelligent technologies are gradually being applied in our daily life, e.g., face recognition [5] and human action recognition [6]. In typical places, like classrooms and shopping malls, pedestrians are easily blocked by other objects, which prevents a precise counting of people. The good news is that this problem happens relatively infrequently on head counting. A computer can be adapted to detect human heads and, in turn, count the number of people. For example, the people counting system can detect the head of the student's heads in the classroom, so that the teachers can know whether a student is absent or not. In another case, the number of people in a self-study room can be counted and fed back to the mobile phone in real-time by head detecting. In this way, the students can quickly know which self-study room still has available seats, avoiding spending lots of time and energy searching for an unoccupied space. Besides these, a shopping mall owner can analyze the laws of customer flow by detecting heads in each store, which helps them make appropriate marketing strategies. All of these demonstrate that high-performance head detection and counting is one of the most crucial techniques in modern AI systems and applications.

2. Related Work

As a fundamental technique of people counting, head counting belongs to target detection in computer vision. A lot of machine learning methods have been proposed for this task. The traditional machine learning target detection algorithms include AdaBoost based on Harr features [7], SVM based on Hog [8] and LBP [9] features, etc. The principle of these detection algorithms mainly depends on the traditional manually extracted features. The procedure usually includes extracting features from the images, then constructing a classifier for classification, and finally obtaining the targets. However, most of these traditional target detection algorithms cannot produce a high accuracy for real applications, neither have a good enough generalization ability.

Deep neural networks, on the other hand, have a much better performance in target detection. Hinton et al. published a deep neural network using RBM coding [10]. Since then, deep learning methods have dominated the implementation of target detection applications. Currently, deep target detection algorithms are mainly divided into three categories. The first one is the multi-stage algorithms such as R-CNN [11] and SPPNet [12]. Then, two-stage implementations like Fast R-CNN [13], Faster R-CNN [14], Mask R-CNN [15], and HyperNet [16] have shown very promising performance. However, the speed of these methods is not fast enough for real applications. Besides these, there are many one-stage algorithms including YoloV1 [17], YoloV2 [18], SSD [19], Retina-Net [20], AlignDet [21], CenterNet [22], FSAF [23], FCOS [24], and YoloV4 [25]. All of the above one-stage algorithms have a fast recognition speed, but the accuracy is far from high enough. There is still a gap to be filled. For this reason, our goal is to improve the YoloV4 model, which represents the current state-of-the-art, to create a high-performance head detection and counting model.

In the conventional YoloV4, non-maximum suppression (NMS) sets the score of adjacent detection frame (adjacent detection frame probably contains object) to 0, then the final output will not contain this detection frame, which caused the occurrence of missed detection. This is harmful in the head-counting application. Soft-NMS algorithm was proposed to attenuate the score of the adjacent detection frame rather than set it to 0 [26]. As long as the score of the adjacent detection frame is greater than the threshold, the final output will contain this detection frame. Inspired by the above inference, this paper proposes a novel head detection method based on YoloV4, which we call Soft-YoloV4 (the NMS in YoloV4 is replaced by Soft-NMS). We make the following novel contributions:

1. We reveal why the conventional YoloV4 model is prone to miss detection in the case of people's heads are blocked by each other;
2. We proposed a new head detection model (Soft-YoloV4) by improving YoloV4. The experiments in two datasets show that the number of people can be counted more accurately by Soft-YoloV4;
3. We compared the Soft-YoloV4 and other methods previously reported, which showed that the Soft-YoloV4 has a better performance and is more conducive to real applications.

The present paper is organized as follows. Section 2 introduces the algorithm design of Soft-YoloV4 and presents the acquisition of experimental data. The results of Soft-YoloV4 in a real application and the comparison of Soft-YoloV4 between other several methods are presented in Section 3. The conclusion is provided in Section 4.

3. Methods

3.1. NMS Algorithm

The YoloV4 model mainly consists of the following parts: CSPDarknet53 (the backbone features extraction network), SPP (the strengthened features extraction network), PANet, and Yolo Head [27]. When the size of the inputted picture is 416 × 416 × 3, the architecture consisting of CSPDarknet53, SPP, PANet, and Yolo Head is shown in Figure 1.

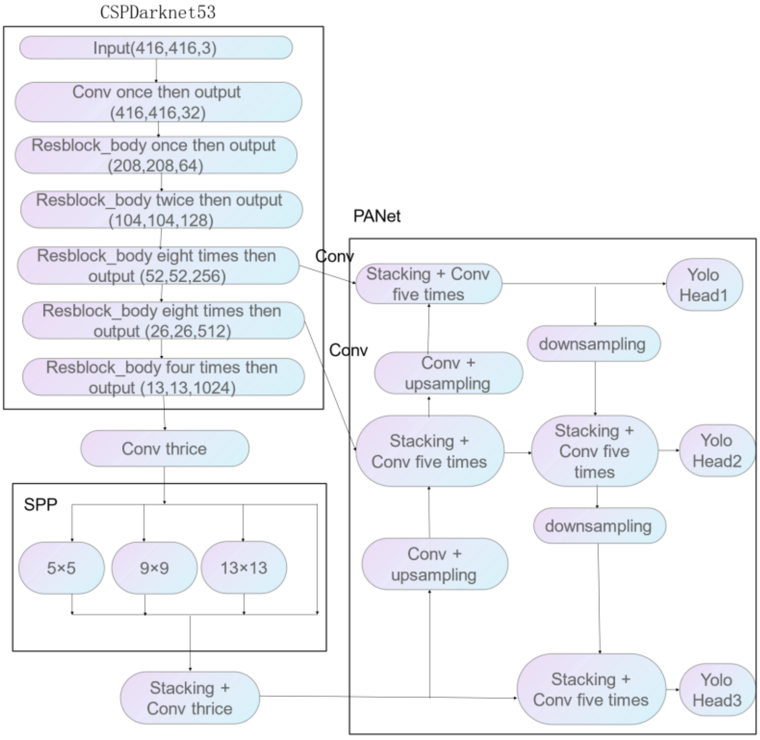

CSPDarknet53
Input(416,416,3)
Conv once then output (416,416,32)
Resblock_body once then output (208,208,64)
Resblock_body twice then output (104,104,128)
Resblock_body eight times then output (52,52,256)
Resblock_body eight times then output (26,26,512)
Resblock_body four times then output (13,13,1024)
Conv thrice
SPP
5×5
9×9
13×13
Stacking + Conv thrice
PANet
Conv
Conv
Stacking + Conv five times
Yolo Head1
downsampling
Conv + upsampling
Stacking + Conv five times
Stacking + Conv five times
Yolo Head2
downsampling
Conv + upsampling
Stacking + Conv five times
Yolo Head3

Figure 1. The architecture of YoloV4 network.

In particular, CSPDarknet53 mainly consists of a series of ResNet [28]. The detailed description can be found in the cspdarknet53 module in Figure 1.

Max-pooling in the SPP architecture mainly uses different pooling kernel sizes of 5 × 5, 9 × 9, 13 × 13. It pools the inputted feature layers and stacks each output. The Max-pooling process reduces the features and parameters of the result and keeps some invariance well, like rotation, translation, expansion, and others. The SPP architecture also increases the receptive field of the output unit nicely.

PANet was proposed by Shu Liu et al. [29]. This architecture makes full use of shallow and deep features. It obtains a more effective feature layer by fusing shallow features and deep features. In YoloV4, PANet is mainly used on three effective feature layers $(13, 13, 1024)$, $(26, 26, 512)$, $(52, 52, 256)$. By fusing the features in PANet, three effective feature layers are available in sizes of $52 \times 52 \times 128$, $26 \times 26 \times 256$, and $13 \times 13 \times 512$, respectively. Yolo Head has two convolution layers: the first layer is a 3×3 convolution, the second is a 1×1 convolution. For the case of Yolo Head1, the input of Yolo Head1 is $52 \times 52 \times 128$ feature layer, and $52 \times 52 \times 18$ feature layer is obtained after Yolo Head1 processing. Likewise, $26 \times 26 \times 18$ feature layer is obtained after Yolo Head2 processing, $13 \times 13 \times 18$ feature layer is obtained after Yolo Head3 processing. Finally, $52 \times 52 \times 18$, $26 \times 26 \times 18$, and $13 \times 13 \times 18$ feature layers will be the output of YoloV4.

In the original YoloV4 model, NMS is used to sift out the detection frame with the highest scores in the same category. However, the elimination mechanism of NMS is very strict, only considering the detection frame and its *IOU* (Intersection over Union), which easily leads to a missed detection. For example, a missed detection as an instance is shown in Figure 2:

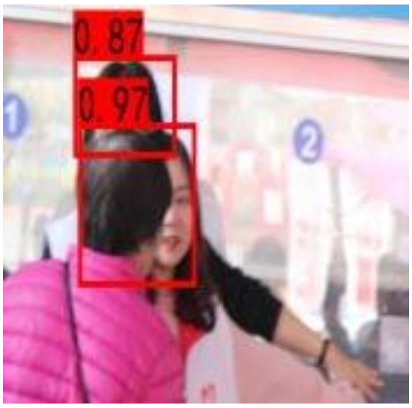


Figure 2. A missed detection happened using NMS.

There are three people in Figure 2. However, only two people were detected using NMS, which means a missed detection. Obviously, in a crowded occasion, using NMS algorithm to remove the redundant detection frames when people's heads are blocked by each other is likely to cause a missed detection.

In our improvement, the key step to achieve people counting is detecting people's heads. When there are too many people, their heads are easily blocked by each other. Therefore, we utilize Soft-NMS to replace NMS in the Soft YoloV4 model to fix the problem. Here, we have the following analysis.

3.2. Principle of Soft-NMS Algorithm

From a mathematical point of view, the mechanism of NMS to remove redundant frames can be expressed as:

$$score_i = \begin{cases} 0, IOU(M, b_i) \geq \text{threshould of } IOU \\ score_i, IOU(M, b_i) < \text{threshould of } IOU \end{cases} \tag{1}$$

where $score_i$ represents the score of the current detection frame. The best threshold of IOU we found is 0.5 after multiple debugging in the data set of this experiment.

In other words, for the detection frame with a higher IOU adjacent to one with the highest score, NMS will set the score of this frame to 0 and then remove it. It is very likely to cause a missed detection when in the situation shown in Figure 2. The mechanism of Soft-NMS to remove redundant detection frames can be expressed as:

$$score_i = score_i e^{-\frac{IOU(M,b_i)^2}{\theta}} \tag{2}$$

It means that Soft-NMS will not directly set the score of the detection frame with a higher IOU adjacent to the one with the highest score to 0. Instead, it penalizes the score. The multiplication of the score of the current detection frame and the weight function is to penalize this detection frame. We used the Gaussian function as the weight function: $e^{-\frac{IOU(M,b_i)^2}{\theta}}$ (θ is the parameter of the weight function. After debugging, the detection effect is the best when θ is 0.1). The higher overlap with the highest-score detection frame, the more severe the score of this detection frame decreases. Finally, only the detection frame with a score higher or equal to 0.5 remains. In this way, Soft-NMS can remove the redundant detection frame and reduce the missed detection rate as well. The flow chart of Soft-NMS is shown in Figure 3.

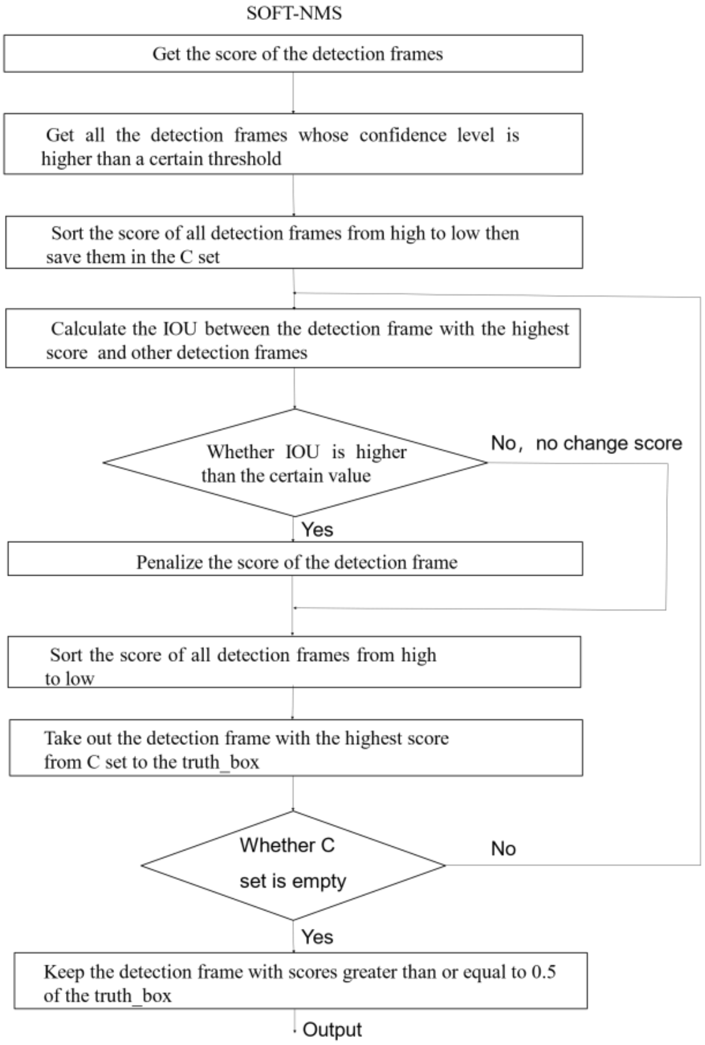

SOFT-NMS
Get the score of the detection frames
Get all the detection frames whose confidence level is higher than a certain threshold
Sort the score of all detection frames from high to low then save them in the C set
Calculate the IOU between the detection frame with the highest score and other detection frames
Whether IOU is higher than the certain value
No，no change score
Yes
Penalize the score of the detection frame
Sort the score of all detection frames from high to low
Take out the detection frame with the highest score from C set to the truth_box
Whether C set is empty
No
Yes
Keep the detection frame with scores greater than or equal to 0.5 of the truth_box
Output

Figure 3. The flow chart of Soft-NMS.

In summary, the main idea of Soft-NMS is as follows. Firstly, it finds out all the detection frames which have a higher confidence level than a certain artificial-set confidence level from an image. The circumstance that the confidence level is lower than this certain confidence level means that there is no target object in the detection frame. Secondly, it processes the detection frames that belong to the same category. Finally, it establishes a set B and puts all the detection frames that belong to the same category into this set. The specific algorithm of Soft-NMS is as follows.

1. Sort the score of the detection frame in set B (this score indicates the probability that the position of the detection frame belongs to this category) from high to low, and choose the frame H with the highest score from the B set.
2. Traverse all the detection frames in set B, and calculate the IOU of each detection frame and the detection frame H with the highest score. Soft-NMS does not directly remove a detection frame from set B but makes a corresponding penalty for this detection frame to decrease the score. The higher the degree of overlapping with the detection frame with the highest score, the more severe the score of this detection frame decreases. Then saving the detection frame H into truth_box.
3. Return to 1. until the set B is empty, and finally, keep the detection frame with a score higher or equal to 0.5 in the truth_box as the output.

After processing Figure 2 by Soft-NMS, the detecting result is as shown in Figure 4.

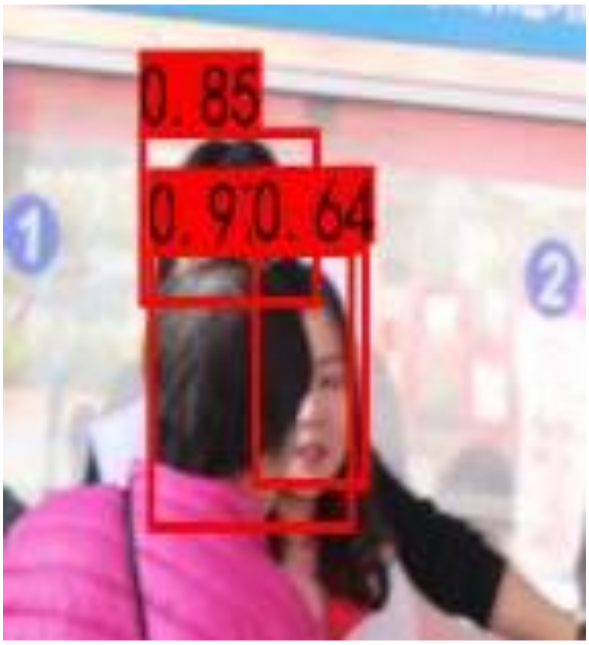


Figure 4. Soft-NMS processing, no missed detection.

4. Experimental Datasets and Evaluation Indexes

The experiments were conducted on two human heads data sets: Brainwash [30] and SCUT_HEAD [31]. The Brainwash data set contains 11,438 images, with a total of 81,975 human heads. The scene in this data set is a coffee shop, and the annotation method of the data set is not the Pascal VOC format. It needs to convert to the Pascal VOC annotation format. The SCUT_HEAD data set contains 4405 images with a total of 11,251 heads. Two data sets include lots of complex scenes, such as classrooms, cafes, daytime, night, and others.

For the case of Brainwash, the size of each image is 640 × 480, 300 images are selected randomly as the testing set, and 11,138 images as the training set. For the case of SCUT_HEAD, the size of each image is different, 141 images are selected randomly as the testing set, and 4264 images as the training set. The third dataset contains all images of A and B, 441 images are selected randomly as the testing set, and 15,402 images as the training set. For the YoloV4 model, the size of the input image is 416 × 416, so all images will be preprocessed, which means all images will be resized to 416 × 416 before being put into the YoloV4 model.

The indexes of the evaluation model in this experiment include the Precision value, the Recall value, and AP value [32]. The calculation of the Precision value and the Recall value are respectively represented by Formulas (3) and (4):

$$\text{Precision} = \frac{TP}{TP + FP} \tag{3}$$

$$\text{Recall} = \frac{TP}{TP + FN} \tag{4}$$

On the above formulas, TP means the prediction result is classified correctly into positive samples, FP indicates the wrong classification into positive samples, and FN represents the wrong into negative samples. The PR curve is the relationship between the Precision value and the Recall value. We can see the PR curve in Figure 5:

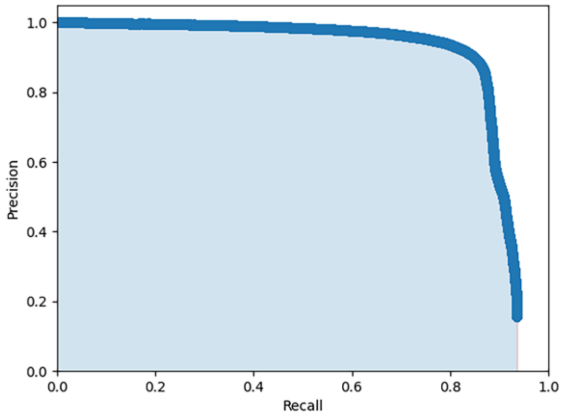


Figure 5. The PR curve.

AP is the area enclosed by the PR curve (the blue area). The higher the value of AP, the better the predictive ability of the model.

5. Results

5.1. Comparison of NMS and Soft-NMS

To verify the efficiency of the Soft YoloV4 model, the same prediction parameters and data sets (more than 400 complex images) are used for head detection in the YoloV4 model using NMS and Soft-NMS. Judging whether the recognition is accurate is based on whether there is a missed detection.

The AP value of the YoloV4 model before improvement is 88.52%, the Precision is 91.15%, and the Recall is 86.93%. When using Soft-NMS, the prediction result of the Soft YoloV4 model is improved, where the AP value is 90.54%, the Precision is 91.94%, and the Recall is 85.55%.

The comparison results on the third dataset between the YoloV4 model before and after improvement are shown in Table 1.

Table 1. The comparison results.

Model	AP/%	Precision/%	Recall/%
Original YoloV4	88.52	91.15	86.93
Soft YoloV4	90.54	91.94	85.55

After contradistinction and analysis, we can see that the AP value and the Precision value are improved compared with the original model. However, the Recall has declined. Soft-NMS remove the redundant detection frame by penalizing the score. There is an adjustable parameter θ in Formula (2). A large parameter θ will result in a smaller penalty, then the redundant detection frame may not be removed, which means the model may indicate that there are two objects although there is only one object. The reason why recall has declined is that the parameter θ is large. Recall or Precision cannot be used to evaluate the effect of the algorithm comprehensively, so the AP index is selected. The experiments proved that the AP value using Soft YoloV4 was higher than that using Original YoloV4, even though recall dropped a little. In this way, replacing NMS with Soft-NMS in YoloV4 is effective.

5.2. Comparison with State-of-the-Arts

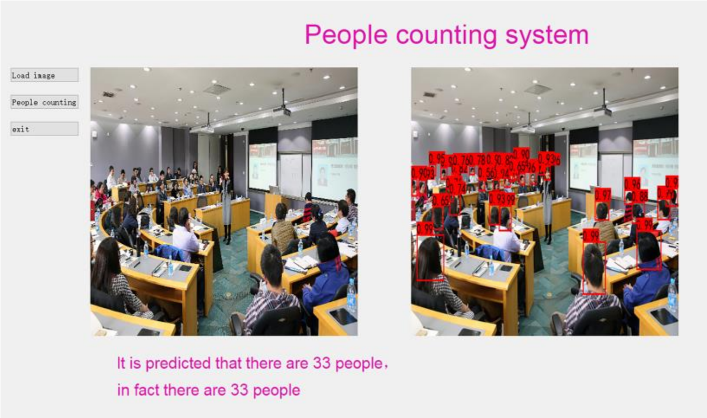
The experiments include the following comparison methods: end-to-end people detection (abbreviated as ReInspect [30]), detecting heads using features refined net and cascaded multi-scale architecture (abbreviated as FRN_CMA [31]), target detection algorithm based on YoloV3 (abbreviated as YoloV3 [33]), and pedestrian head detection algorithm based on clustering and Faster RCNN (abbreviated as CFR-PHD [34]). All methods use the same evaluation index. The detection results of each method on the Brainwash data set and SCUT_HEAD data set are shown in Table 2.

Table 2. Experimental results obtained on Brainwash and SCUT_HEAD.

Methods	Brainwash (AP/%)	SCUT_HEAD (AP/%)
ReInspect	78.10	77.50
FRN_CMA	88.10	86.30
YoloV3	85.11	84.13
CFR-PHD	90.20	87.70
Soft YoloV4	92.29	91.70

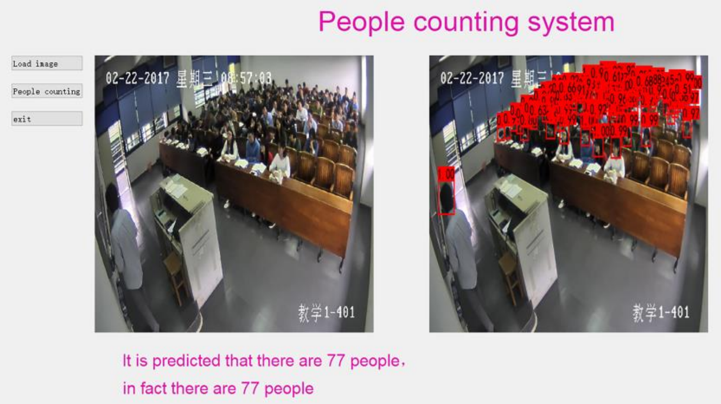
According to the experiment results on the Brainwash data set and the SCUT_HEAD data set, our Soft YoloV4 algorithm improves detection performance compared to the above algorithms. On the Brainwash data set, the AP value dramatically increases. Compared to the ReInspect, FRN_CMA, YoloV3, and CFR-PHD algorithms, the improvements are 14.19%, 4.19%, 7.18%, and 2.09%, respectively. On the SCUT_HEAD data set, the improvements by the AP value are 14.20, 5.40, 7.57, and 4.00%. Therefore, the performance of our proposed improvement can be approved.

Here are three examples, as shown in the following Figures 6–8.



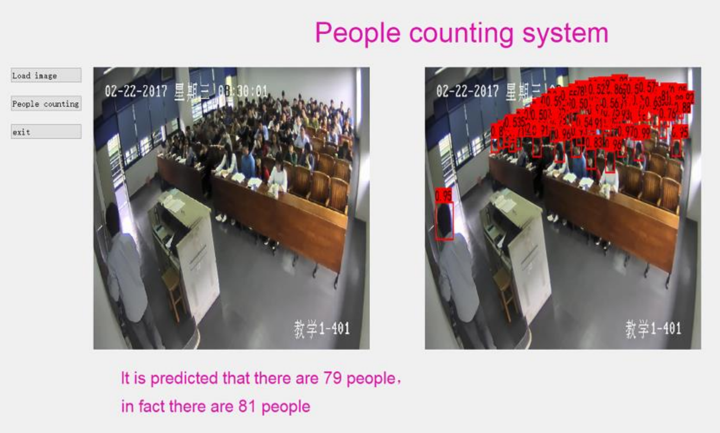
It is predicted that there are 33 people，
in fact there are 33 people

Figure 6. One example of people counting results. There are 33 people in the classroom, and it was predicted that there would be 33 people. The result is completely correct.



It is predicted that there are 77 people，
in fact there are 77 people

Figure 7. One example of people counting results. There are 77 people in the classroom, and it was predicted that there would be 77 people. The result is completely correct.


People counting system

It is predicted that there are 79 people，
in fact there are 81 people

Figure 8. One example of people counting results. There are 79 people in the classroom, and it was predicted that there would be 81 people. The result is not completely correct.

The result in Figure 8 is not completely correct. With the increase of pedestrian density in a scene, the visibility of heads decreases with the increase of mutual occlusions, resulting in the decrease of head detection, as shown in Figure 8. The possible reason why the model cannot predict objects over heavily overlapped with others is that a detection frame only predicts an object rather than a set of correlated objects.

6. Conclusions

Compared with other target detection models, the Soft-YoloV4 model in this paper has a higher recognition accuracy and a better people counting effect. Soft-YoloV4 can be built on the server. By recognizing the images sent by the client, the server can return the specific number of people to the client. In this way, the number of people in the classroom can be counted conveniently and quickly, which helps teachers count the number of students, and students do not need to go to each classroom to check whether there is an available seat for them, and then quickly choose a self-study room.

This paper is still unable to accurately recognize the situation that the degree of blockage is too high. In the future, we can consider combining the human body model to determine whether there is a blockage in the detection frame. The network architecture of the target detection model is also too large. Although the accuracy is high, the detecting speed is relatively slow. The next step is to modify the network architecture of the model to speed up the recognition process without significantly decreasing the accuracy. KuralNet is a lightweight deep learning model that strikes a good balance between parameters and effectiveness [35]. In the KuralNet, the inverse residual block with deep convolution and frequency-doubling convolution can be used for signal processing to reduce the computational cost. Perhaps we can learn from this to reduce the complexity of Soft-YoloV4.

This paper proposes a head detection model by improving YoloV4 to count the number of people. By detecting people's heads, we have an improved version YoloV4 using Soft-NMS. In this way, the number of people can be counted more accurately and performance close to the requirement of real applications is obtained. The original YoloV4 model uses the NMS algorithm to remove redundant detection frames. The Soft-YoloV4 model uses the Soft-NMS algorithm. After comparative analysis, Soft-YoloV4 has a higher accuracy in head detection. The AP value of Soft-YoloV4 is 90.54%, 2.02% higher than the original YoloV4 model. Therefore, Soft-YoloV4 is more suitable for head detection on crowded occasions.

Author Contributions: Conceptualization, Z.Z. and S.Z.; Data curation, S.X.; Formal analysis, Z.Z., S.X., Y.C. and C.Y.; Funding acquisition, Z.Z.; Investigation, S.X.; Supervision, S.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by [Young innovative talents project of colleges and universities in Guangdong Province] grant number [2021KQNCX092]; [Doctoral program of Huizhou University] grant number [2020JB028]; [Outstanding youth cultivation project of Huizhou University] grant number [HZU202009].

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository that does not issue DOIs. Brainwash dataset and SCUT_HEAD dataset were analyzed in this study. Brainwash dataset can be found here [https://github.com/aditya-vora/FCHD-Fully-Convolutional-Head-Detector] (accessed on 30 November 2021). SCUT_HEAD dataset can be found here [https://github.com/HCIILAB/SCUT-HEAD-Dataset-Release] (accessed on 30 November 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Tan, Y.L. Statistical Image Recognition Algorithm Based on Skin Color. *J. Huaihai Inst. Technol.* **2014**, *23*, 36–39.
2. Zhang, L. *Population Density Statistics Based on Face Detection*; Lanzhou University of Technology: Lanzhou, China, 2018.
3. Jin, Y.H. *Video Pedestrian Detection and People Counting*; Inner Mongolia University: Hohhot, China, 2018.
4. Zeng, S.; Zhang, B.; Gou, J. Learning double weights via data augmentation for robust sparse and collaborative representation-based classification. *Multimed. Tools Appl.* **2020**, *79*, 20617–20638. [CrossRef]
5. Rathgeb, C.; Dantcheva, A.; Busch, C. Impact and detection of facial beautification in face recognition: An overview. *IEEE Access* **2019**, *7*, 152667–152678. [CrossRef]
6. Li, W.; Nie, W.; Su, Y. Human action recognition based on selected spatio-temporal features via bidirectional LSTM. *IEEE Access* **2018**, *6*, 44211–44220. [CrossRef]
7. Zhang, C.L.; Liu, G.W.; Zhan, X.; Cai, H.; Liu, Z. Face detection algorithm based on new haar features and improved AdaBoost. *J. Chang. Univ. Sci. Technol. (Nat. Sci. Ed.)* **2020**, *43*, 89–93.
8. Tan, G.X.; Sun, C.M.; Wang, J.H. Design of video vehicle detection system based on HOG features and SVM. *J. Guangxi Univ. Sci. Technol.* **2021**, *32*, 19–23, 30.
9. Gu, W. Research on moving target detection algorithm based on LBP texture feature. *Off. Informatiz.* **2017**, *22*, 21–24.
10. Hinton, G.E.; Salakhutdinov, R.R. Reducing the Dimensionality of Data with Neural Networks. *Science* **2006**, *313*, 504–507. [CrossRef] [PubMed]
11. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
12. He, K.; Zhang, X.; Ren, S.; Sun, J. Spatial pyramid pooling in deep convolutional networks for visual recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *37*, 1904–1916. [CrossRef] [PubMed]
13. Li, J.; Liang, X.; Shen, S.; Xu, T.; Feng, J.; Yan, S. Scale-aware fast R-CNN for pedestrian detection. *IEEE Trans. Multimed.* **2017**, *20*, 985–996. [CrossRef]
14. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Trans. Pattern Anal. Mach. Intell.* **2016**, *39*, 1137–1149. [CrossRef] [PubMed]
15. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2961–2969.
16. Kong, T.; Yao, A.; Chen, Y.; Sun, F. Hypernet: Towards accurate region proposal generation and joint object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 845–853.
17. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 779–788.
18. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
19. Liu, W.; Anguelov, D.; Erhan, D.; Szegedy, C.; Reed, S.; Fu, C.Y.; Berg, A.C. Ssd: Single shot multibox detector. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 11–14 October 2016; pp. 21–37.
20. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
21. Chen, Y.; Han, C.; Wang, N.; Zhang, Z. Revisiting feature alignment for one-stage object detection. *arXiv* **2019**, arXiv:1908.01570.
22. Duan, K.; Bai, S.; Xie, L.; Qi, H.; Huang, Q.; Tian, Q. Centernet: Keypoint triplets for object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 6569–6578.
23. Zhu, C.; He, Y.; Savvides, M. Feature selective anchor-free module for single-shot object detection. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 840–849.

24. Tian, Z.; Shen, C.; Chen, H.; He, T. Fcos: Fully convolutional one-stage object detection. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Korea, 27–28 October 2019; pp. 9627–9636.
25. Bochkovskiy, A.; Wang, C.Y.; Liao, H.Y.M. Yolov4: Optimal speed and accuracy of object detection. *arXiv* **2020**, arXiv:2004.10934.
26. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-NMS—Improving object detection with one line of code. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 5561–5569.
27. Neubeck, A.; van Gool, L. Efficient non-maximum suppression. In Proceedings of the 18th International Conference on Pattern Recognition (ICPR'06), Hong Kong, China, 20–24 August 2006; Volume 3, pp. 850–855.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
29. Liu, S.; Qi, L.; Qin, H.; Shi, J.; Jia, J. Path aggregation network for instance segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–22 June 2018; pp. 8759–8768.
30. Stewart, R.; Andriluka, M.; Ng, A.Y. End-to-end people detection in crowded scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 2325–2333.
31. Peng, D.; Sun, Z.; Chen, Z.; Cai, Z.; Xie, L.; Jin, L. Detecting heads using feature refine net and cascaded multi-scale architecture. In Proceedings of the 2018 24th International Conference on Pattern Recognition (ICPR), Beijing, China, 20–24 August 2018; pp. 2528–2533.
32. Zheng, L.; Shen, L.; Tian, L.; Wang, S.; Wang, J.; Tian, Q. Scalable person re-identification: A benchmark. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 1116–1124.
33. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
34. Zhang, J.; Chen, L.; Li, Z.; Wang, S.; Chen, Z. Pedestrian head detection algorithm based on clustering and Fast RCNN. *J. Northwest Univ.* **2020**, *50*, 971–978.
35. Ayala, A.; Fernandes, B.; Cruz, F.; Macêdo, D.; Oliveira, A.L.; Zanchettin, C. KutralNet: A portable deep learning model for fire recognition. In Proceedings of the 2020 International Joint Conference on Neural Networks (IJCNN), Glasgow, UK, 19–24 July 2020; pp. 1–8.