applied sciences

MDPI

Article

Pedestrian Flow Tracking and Statistics of Monocular Camera Based on Convolutional Neural Network and Kalman Filter

Miao He [1,2,3,4,5,*], Haibo Luo [1,2,4,5], Bin Hui [1,2,4,5] and Zheng Chang [1,2,4,5]

1 Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China; luohb@sia.cn (H.L.); huibin@sia.cn (B.H.); changzheng@sia.cn (Z.C.)
2 Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China
3 University of Chinese Academy of Sciences, Beijing 100049, China
4 Key Laboratory of Opto-Electronic Information Processing, Chinese Academy of Science, Shenyang 110016, China
5 The Key Lab of Image Understanding and Computer Vision, Shenyang 110016, China
* Correspondence: hemiao@sia.cn; Tel.: +86-024-2397-0757

Received: 19 March 2019; Accepted: 15 April 2019; Published: 18 April 2019

Abstract: Pedestrian flow statistics and analysis in public places is an important means to ensure urban safety. However, in recent years, a video-based pedestrian flow statistics algorithm mainly relies on binocular vision or a vertical downward camera, which has serious limitations on the application scene and counting area, and cannot make use of the large number of monocular cameras in the city. To solve this problem, we propose a pedestrian flow statistics algorithm based on monocular camera. Firstly, a convolution neural network is used to detect the pedestrian targets. Then, with a Kalman filter, the motion models for the targets are established. Based on these motion models, data association algorithm completes target tracking. Finally, the pedestrian flow is counted by the pedestrian counting method based on virtual blocks. The algorithm is tested on real scenes and public data sets. The experimental results show that the algorithm has high accuracy and strong real-time performance, which verifies the reliability of the algorithm.

Keywords: pedestrian flow statistics; neural network; Kalman filter; multi-object tracking; data association

1. Introduction

Pedestrian flow statistics is an important application in the field of computer vision [1]. It is a key technology in intelligent cities, intelligent retail, public place security and many other fields [2]. In recent years, with the continuous progress of intelligent city, pedestrian flow statistics has attracted more and more researchers and companies to participate in, and developed more and more statistics algorithms [3,4].

The earliest methods for pedestrian traffic statistics depend on manual statistics or bill statistics, which are either costly or have an impact on pedestrians. Then pedestrian flow statistics methods based on pressure sensor or photoelectric sensor are proposed. However, the accuracy of these methods are not good enough for dense pedestrian flow with severe occlusion. Due to the development of computer vision, pedestrian traffic statistics methods based on vertical downward stereo camera have emerged [5]. This kind of method is the most popular pedestrian traffic statistics method at present. Because the vertical downward camera can effectively avoid pedestrian occlusion, and the binocular-vision-based three-dimensional reconstruction algorithm can well filter out complex color

background, the method has high accuracy, but the installation location and visual field are severely limited. It can only be applied to narrow indoor entrances, but not to commercial streets with wide outdoor entrances. At the same time, it is impossible to use the large number of front-down monocular cameras in the city.

Compared with vertical downward stereo camera, front-down monocular camera has wider vision, and because it can see the positive human face, it can better take into account security, criminal investigation and abnormal behavior early warning tasks. In front-down cameras, the image plane is about 45 degrees from the ground. The occlusion between the targets in front-down cameras is not as serious as that in front cameras. At the same time, the scale difference between targets is relatively small. There is a negative correlation between the size of the pedestrian target and the placement height of the camera.

However, due to the challenges of frequent occlusions, illumination changes, target scale changes, different fog concentrations in different distance and so on, pedestrian flow statistics algorithm based on front-down monocular camera puts forward higher requirements for detection and tracking algorithm. Thanks to the rapid progress of deep learning algorithm [6–11], the accuracy of pedestrian detection is constantly improving. More accurate detection results make the performance of tracking-by-detection method reach a higher level [12], which makes it possible to develop a high-precision pedestrian flow statistics algorithm based on front-down monocular camera.

In this paper, we propose a pedestrian flow tracking and statistics method based on front-down monocular camera. Firstly, we use convolutional neural network to detect pedestrians appearing in the camera, and modify the detection results based on intersection over union and aspect ratio. Secondly, we use Kalman filter to build uniform linear motion models for the detected pedestrians. Then pedestrian tracking is accomplished by data association algorithm. Finally, the virtual block method is used to count the target. We test the proposed algorithm using the real scenic spot entrance surveillance video, the F1 score of pedestrian flow statistics has reached 95%. At the same time, we compare the proposed multi-target tracking algorithm with other multi-target tracking algorithms on the 2DMOT2015 dataset. The MOTA of the proposed algorithm reaches 48.1, and the algorithm has advantages in computing speed.

The rest of this paper is organized as follows. Section 2 reviews the related work. The proposed algorithm is described in detail in Section 3. In Section 4, experiments and comparisons are carried out. Conclusion and analysis are presented in Section 5.

2. Related Work

The pedestrian flow statistics algorithm mainly includes three steps: pedestrian detection, multi-target tracking and pedestrian counting.

Current fast pedestrian detection methods are mainly divided into two categories. The first is pedestrian detection method based on background modeling, which mainly relies on the background modeling method to extract the foreground moving object, and uses classifier to judge whether the moving object is a pedestrian. GMM algorithm [13] and vibe algorithm [14] are the most representative. This kind of algorithm is fast in speed, but it cannot cope with the change of illumination and the jitter of camera very well. At the same time, it is difficult to distinguish dense objects or objects that are occluded from each other. The second kind of pedestrian detection algorithm is based on statistical learning. The algorithm has high accuracy and can cope with occlusion and environmental changes to a certain extent. The HOG + SVM [15] proposed by Dalal et al. is the most classical algorithm of this kind. Because of the popularity of deep learning methods in recent years, the accuracy of pedestrian detection algorithm based on convolutional neural network has reached an unprecedented height [16]. Detection methods based on deep learning can be divided into two-stage method and one-stage method. Faster R-CNN proposed by Shaoqing Ren [17] and its subsequent variants [18,19] belong to the two-stage method, which has advantages in detection and positioning accuracy. WeiLiu et al.'s

SSD detection algorithm [20] and its subsequent variants [21,22] belong to one-stage algorithm, which achieves good accuracy and better real-time performance.

Multi-objective tracking methods can be divided into DBT (detection-based-tracking) methods and DFT (detection-free-tracking) methods based on initialization method. DFT methods needs to label the targets manually, and then track them in subsequent frames [23,24]. DBT methods completes tracking by detecting the targets in each frame and putting the targets into the tracklets [25–27]. DBT methods are more suitable for pedestrian traffic statistics applications because of the frequent appearance of new targets and the frequent disappearance of old ones. At the same time, according to the processing mode, multi-target tracking methods can be divided into online and offline algorithms. Online algorithm only uses the current frame of image sequence and several previous frames [28,29], which is more suitable for applications that need real-time implementation. The offline algorithm needs to use some future frames in the image sequence [30,31], which is more suitable for post-analysis and processing of video. Appearance model is widely used in the field of multi-target tracking. The appearance model has an important role in associating tracklets and detections. With the help of the appearance model, the ID scitches can be effectively suppressed. Ullah, M. et al. proposed a multi-target tracking method establishing appearance model with HoG descriptor [32]. Bae, S.H. et al. proposed a deep appearance learning method to learn a discriminative appearance model which can distinguish multiple objects with large appearance variations [33]. At the same time, the motion model has the same important role as the appearance model. Since the motion of the target in the image is usually relatively flat, the estimation of the trend of the target motion can predict the position of the target in the next frame, thus reducing the search area and even directly obtaining the tracking results.

The pedestrian counting method was originally road marking method. In this method, a mark is set on the road surface, and when the mark is covered, a pedestrian is judged to pass. Then Kryjak et al. proposed a counting method based on virtual lines [34]. The main idea is that when the target center passes through the virtual line, a pedestrian is judged to pass through. However, if there is a target hovering near the virtual line, it will seriously affect the counting accuracy. Later Xu et al. proposed a counting method based on double virtual lines [2]. In this method, two virtual lines are delineated, and the sequence of pedestrians passing through the virtual lines is judged to realize the counting.

3. Methodology

The proposed algorithm can be divided into three parts: pedestrian detection, multi-pedestrian tracking and pedestrian counting. The overall flow of the proposed method is shown in Figure 1.

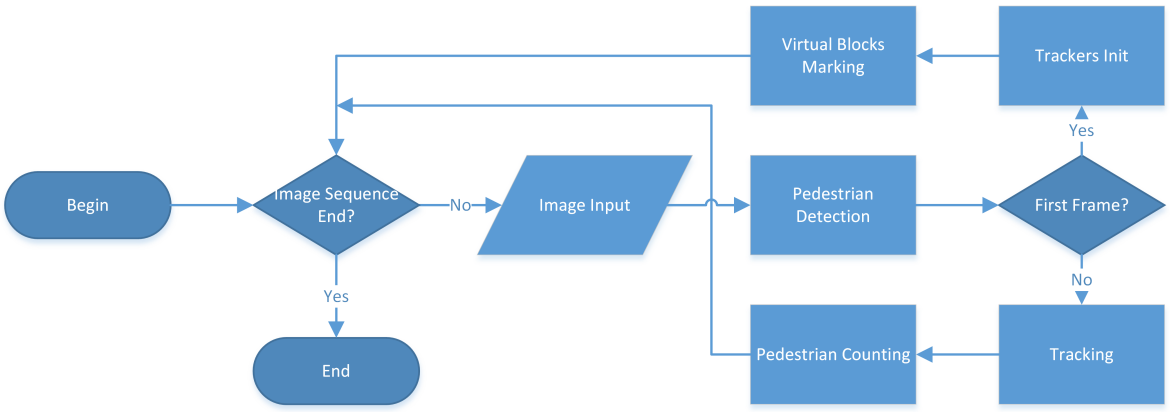

Figure 1. Overall flow chart of the proposed algorithm.

3.1. Pedestrian Detection

Pedestrian detection is the first step of pedestrian flow statistics algorithm. The algorithm mainly improves from the yolov3 detection network [35]. In order to reduce the computational complexity

of the algorithm, the darkent-53 network in the front of the network is replaced by a pruned and compressed VGG network [36], which reduces the computational complexity of the network from 65 Bflops to 39 Bflops. Because the target is pedestrian traffic statistics, too small anchor settings have no effect on improving detection accuracy, so kmeans algorithm is used to re-cluster the size of network anchor. Due to the large scale of the targets in the front-down camera, the last FPN structure of the yolov3 network is removed to reduce the computational complexity. The final computational complexity of the neural network is 34 Bflops. At the same time, the mAP (mean Average Precision) of the detector only reduce 1.61, from 74.49 to 72.88. The final network structure of the algorithm is shown in Figure 2.

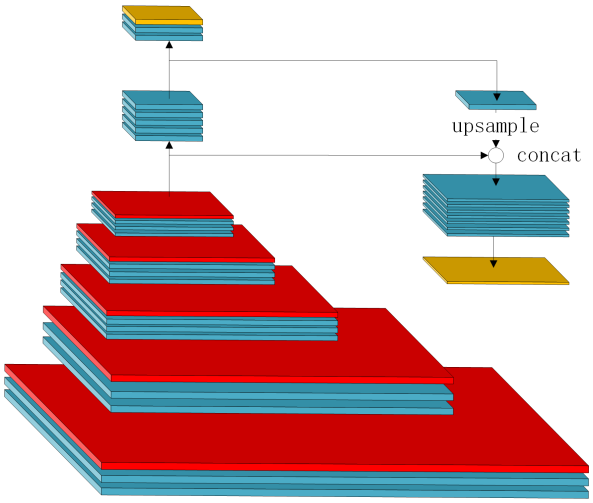

Figure 2. Network structure of detection algorithms. The blue layers are convolutional layers, the red layers are max-pooling layers, and the yellow layers are detection layers.

In pedestrian flow statistics, pedestrian targets are very dense, and there is serious occlusion between the targets. The traditional non-maximum suppression (NMS) method can cause a part of the correct detection to be lost while removing redundant detection bounding boxes. So soft-NMS method is used to improve the non-maximum suppression [37]. Unlike the original non-maximum suppression method, as shown in Formula (1), soft-mns method does not directly remove the bounding box whose IOU exceeds the threshold and confidence is lower, but reduces the confidence of the detection box, which makes it more difficult for the correct detection to be removed incorrectly due to the dense targets. In Formula (1), d_i is a detection result with score s_i, d_m is another detection result which has higher score than d_i, N_i represent the threshold of soft-NMS. As shown in Figure 3, after using soft-NMS, not only the redundant detection results can be correctly removed, but also the missing rate can be reduced.

$$s_i = \begin{cases} s_i, & IOU(d_m, d_i) < N_i \\ s_i(1 - IOU(d_m, d_i)), & IOU(d_m, d_i) \geq N_i \end{cases} \quad (1)$$

In the front-down cameras, the objects close to the cameras can easily occlude the lower half of the objects farther from the cameras. On this basis, because of the lateral movement of the target, the bounding box of the farther target will change dramatically in height, which is not conducive to the following tracking operation. However, the top position and width of the bounding box will not be affected in such case. Based on this observation, we adjust the shape of bounding boxes by their width and the top position. For the bounding boxes whose aspect ratio are greater than 1:2.5, the heights of them are increased on the basis of fixing the top position and width, so that the aspect ratios are adjusted to 1:2.5. As shown in Figure 4, this scheme effectively reduces the deformation of the bounding box due to occlusion. At the same time, this method improves the positioning accuracy of the real center position of the occluded target, and is more conducive to the final counting task.

Figure 3. Detection result with soft-NMS (left) and NMS (right). The bounding box for the person in yellow is not miss deleted as redundant detection with soft-NMS.

Figure 4. The deformation of the bounding box of a single target without (**left**) and with (**right**) shape adjusting.

3.2. Multi- Pedestrian Tracking

The tracking algorithm is a multi-target tracking method based on detection results. The tracking method can be divided into two parts. Firstly, a motion model is built for the detected target, and new tracklets are built for new targets. The second part is the data association algorithm, which matches the target detected in each frame with the existing tracklets by the cost function to achieve the purpose of detection. The overall flow of the tracking algorithm is shown in Figure 5.

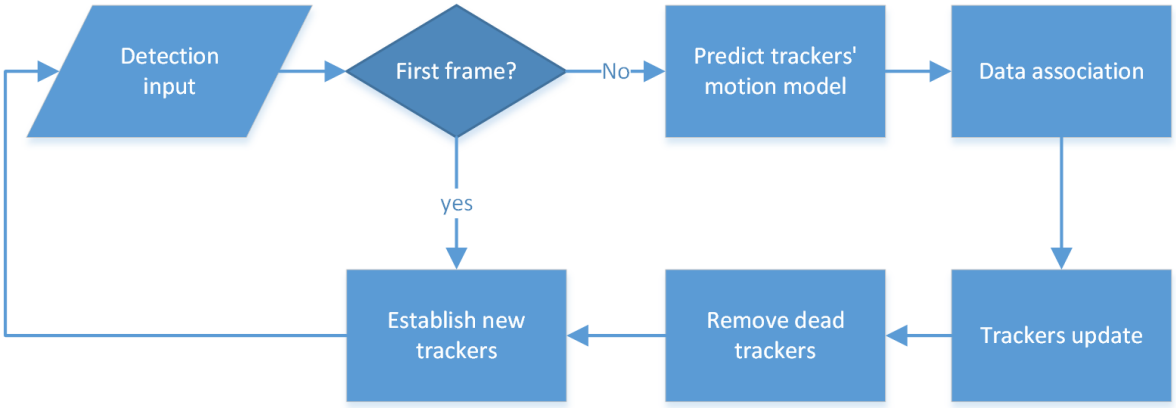

Figure 5. Overall flow of the tracking algorithm.

In the tracking algorithm, the linear motion model is chosen as the motion model of the tracklet. The model is based on Kalman filtering algorithm, and the target state is expressed as

$[u, v, s, r, u', v', s']^T$. Where (u, v) is the coordinate of the center position of the bounding box, and (s, r) are the scale and aspect ratio of the bounding box respectively. (u', v') is the speed of the target in horizontal and vertical directions, s' represents the changing rate of the scale of the target. Since the aspect ratio of the bounding box is adjusted before, it is assumed that the aspect ratio of the target does not change here.

Data association algorithm uses Hungarian algorithm to match existing tracklets and detection results in current frame. The cost function is divided into three parts: IOU limit, scale changing limit and standardized distance. It is required that the IOU of the tracklet's state and the detection result is larger than a certain threshold, and the scale change is lower than a certain threshold, otherwise the tracklet and the detection result will not match each other. Scale changes are described by the following formula:

$$D_{scale} = min(\frac{max(w_1, w_2)}{min(w_1, w_2)}, \frac{max(h_1, h_2)}{min(h_1, h_2)}) - 1 \quad (2)$$

where w_1 and h_1 are the width and height of the detection result respectively, w_2 and h_2 are the width and height of the tracklet state respectively.

On the basis of these two limitations, the matching degree between a detection result and a tracklet is mainly described by standardized distance. The standardized distance standardizes the pixel distance between the detector and the tracklet by the minimum width and height of them, which can effectively reduce the difference caused by the depth of field between the pixel distance and the actual distance. The standardized distance is shown as follows:

$$D_{sp} = \sqrt{(\frac{x_1 - x_2}{min(w_1, w_2)})^2 + (\frac{y_1 - y_2}{min(h_1, h_2)})^2)} \quad (3)$$

where (x_1, y_1) is the coordinate of the detection result, (x_2, y_2) is the coordinate of the tracklet state.

For the successfully matched tracklets and detections, the status of the tracklets are updated by the positions of the detections. In this case, the state of the tracklet, including position, scale, velocity and scale change rate, is the optimal estimate obtained by Kalman filter. Unmatched detectors are candidates for new targets and candidate tracklets for them are established. If these candidate tracklets match detection results in consecutive multiple frames, they will be used as new tracklets for targets newly appear in these frames. Motion models established by detection results in the first frame of the image sequence are used as new tracklets immediately for the initialization of the sequence. The unmatched tracklet outputs the predicted results directly. In this case, the tracking state is completely determined by the prediction matrix, and the predicted tracklet moves in a straight line with a uniform speed decided by the state variables. This method can reduce the influence caused by occlusion or detector failure in a short time. When a tracklet fails to match any detection result in continuous multiple frames, it is considered that the target tracked by the tracklet has disappeared and the tracklet is deleted.

3.3. Pedestrian Counting

A pedestrian counting algorithm based on virtual blocks is proposed. Similar to the counting method based on double virtual lines, the pedestrian counting algorithm based on virtual blocks counts the number of pedestrians according to the sequence of blocks passed by the pedestrian detection center. This method shortens the time requirement of continuous target tracking and makes the holistic algorithm more robust to occlusion.

Block-based counting algorithms need to delimit the beginning area, count area and end area, as shown in Figure 6. If the center of the target is initialized in the beginning area and reaches the end area after passing through the counting area, the count is made once. Two-way counting can be realized by delimiting regions in different order.

Compared with the counting method based on double virtual lines, the block-based method can better adapt to the counting area with different shapes, which is more suitable for the application scenarios of front-down cameras. In addition, the block-based method is easier to achieve the effect of counting part of the road area through flexible setting of counting area. In addition, by setting the start and end areas, it is easier to count only for the target entering from a specific entry or leaving from a specific exit.

Figure 6. Virtual block-based method, pedestrian counting method.

4. Experiment

4.1. The Performance of Pedestrian Flow Statistics Algorithms in Real Scene

Since there is no dataset specifically for pedestrian flow statistics, we use the image sequence captured by front-down cameras at the entrance of crowded scenic spots to verify the detection and counting effect of the proposed algorithm.

To develop an algorithm with high robustness against the changes of illumination and fog concentration, we have manually detected and labeled the images captured by the camera, and made a new dataset. As a detection dataset, five half-hour video recordings were collected. There were intense changes in illumination in the videos, and some videos have dense fog. The frame rate of video recordings is 25 fps, and an image is saved every 12 frames. The total number of annotated images is 18,750, including 282,674 pedestrian bounding boxes. Among them, 2000 images from the end of two videos with different perspectives are used as the test set, while the remaining 16,750 images are used as the training set. After training, the detection effect of the algorithm is evaluated, and the most popular evaluation indicators in the detection field, such as precision, recall, mAP (mean Average Precision), are selected as the evaluation indicators. The detection performance is shown in Table 1. Compared with detectors trained only with coco datasets, the robustness of the detector trained with new images to the changes of illumination and fog concentration is greatly improved.

Table 1. Detection performance of detector trained with different data.

	Precision↑	Recall↑	F1-Score↑	TP↑	FP↓	FN↓	mAP↑
detector (coco)	0.87	0.67	0.76	17786	2644	8887	72.88%
detector (coco + new data)	0.86	0.79	0.82	21156	3550	5517	83.39%

For pedestrian counting effect, we selected five other videos to evaluate. By observing the counting results manually, the missing alarm and false alarm of the counting algorithm are counted. Finally, the counting algorithm is evaluated by the precision, recall and F1-Score of the counting algorithm. The evaluation results of the algorithm in five videos are shown in Table 2.

Table 2. The evaluation results of the algorithm in test videos.

Sequence	Count Result	GT	FP	FN	TP	Precision	Recall	F1-Score
Seq1	552	567	21	36	531	96.20%	93.65%	94.91%
Seq2	411	412	18	19	393	95.62%	95.39%	95.50%
Seq3	571	572	21	22	550	96.32%	96.15%	96.24%
Seq4	923	955	18	50	905	98.05%	94.76%	96.38%
Seq5	669	693	16	40	653	97.61%	94.23%	95.89%
TOTAL	3126	3199	94	167	3032	96.99%	94.78%	95.87%

It can be seen that the mean F1-score of the algorithm is over 95% in the five videos, which proves the effectiveness of the algorithm. The running speed of the algorithm on NVIDIA GTX1060 GPU is 28.4 FPS, which has good real-time performance. The actual running effect is shown in Figure 7.

Figure 7. The actual operation effect of the algorithm.

4.2. The Performance of Tracking-by-Detection Algorithm Compared with Other Algorithms

In pedestrian flow statistics algorithms, the performance of detecting and tracking algorithms has an important impact on the counting results. Therefore, we evaluate the proposed algorithm by comparing the comprehensive performance of the proposed detecting and tracking algorithm with other algorithms.

Here we choose to use the 2DMOT2015 benchmarks [38] to evaluate the performance of the algorithm quantitatively. 2DMOT2015 benchmark is a well-known framework for the fair evaluation of multi-pedestrian tracking algorithms. The dataset includes 22 image sequences, of which 11 are training sets and 11 are test sets. Image sequences come from several influential pedestrian detection and tracking datasets, including KITTI, ADL, ETH, PETS and TUD.

The main evaluation indicators are as in Table 3.

Table 3. Main evaluation indicators.

Measure	Better	Description
MOTA	higher	Multiple Object Tracking Accuracy.
MOTP	higher	Multiple Object Tracking Precision.
IDF1	higher	The ratio of correctly identified detections.
MT	higher	Mostly tracked targets.
ML	lower	Mostly lost targets.
FP	lower	The total number of false positives.
FN	lower	The total number of false negatives.
ID Sw.	lower	The total number of identity switches.

Among them, MOTA is a comprehensive evaluation of FP, FN and IDSW. Its formula is as follows:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FN}_t + \text{FP}_t + \text{IDSW}_t)}{\sum_t GT_t} \tag{4}$$

where GT_t is the number of ground truth in frame t.

MOTP focuses on the average difference between TP and its corresponding ground truth. The formula is as follows:

$$\text{MOTP} = \frac{\sum_{t,i} d_{t,i}}{\sum_t c_t} \tag{5}$$

Where $d_{t,i}$ denotes the overlap of the bounding box i with its corresponding ground truth, c_t denotes the amount of bounding boxes which match ground truths successfully.

To verify the comprehensive performance of detector and tracker, we compare the proposed algorithm with other algorithms using private detector, including MDP_Subcnn [39], DMT [40] and Sort [41]. The specific comparison results are shown in Table 4.

Table 4. The performance of proposed tracking-by-detection algorithm compared with other algorithms in 2DMOT2015 dataset. The green and yellow colors indicate the best and the second best algorithm in each measure.

Sequence	Algorithm	MOTA	IDF1	MOTP	MT	ML	FP	FN	ID Sw
TUD-Crossing	MDP_Subcnn	78.9	74.5	76.7	0.692	0	32	195	6
	DMT	70	56.3	73.3	0.615	0	73	229	29
	Sort	67.5	57.6	74.5	0.462	0.077	32	308	18
	PFS(our)	74.6	49.9	76.2	0.615	0	49	193	38
PETS09-S2L2	MDP_Subcnn	47.5	36.8	72.6	0.071	0.095	341	4524	196
	DMT	47.7	36.8	70.4	0.214	0.19	502	4429	113
	Sort	27.4	23.6	67.4	0	0.262	806	5958	240
	PFS(our)	56.7	32	74.4	0.19	0.048	358	3519	295
ETH-Jelmoli	MDP_Subcnn	48.2	65.7	77.3	0.356	0.222	492	814	9
	DMT	48.2	62.8	75	0.6	0.111	758	529	26
	Sort	39	52.9	74.1	0.2	0.289	439	1071	38
	PFS(our)	61.5	59	76.2	0.356	0.267	176	754	46
ETH-Linthescher	MDP_Subcnn	63.9	67.1	77.1	0.244	0.31	495	2657	70
	DMT	60.5	60.2	76.4	0.437	0.244	1425	1963	138
	Sort	52.2	54.5	73.8	0.147	0.406	397	3725	144
	PFS(our)	62.8	57.5	76.8	0.335	0.223	735	2374	215
ETH-Crossing	MDP_Subcnn	63.8	76.5	79.5	0.192	0.269	64	293	6
	DMT	59	54.7	81.5	0.308	0.308	169	221	21
	Sort	55.4	49.7	80.3	0.154	0.385	58	368	21
	PFS(our)	62.4	53.9	79.5	0.308	0.077	135	215	27
AVG-TownCentre	MDP_Subcnn	49.5	64.5	70.1	0.389	0.155	1381	2106	121
	DMT	45.5	54.6	68.5	0.283	0.274	653	3127	117
	Sort	27.2	45.1	67.4	0.058	0.279	1111	3930	162
	PFS(our)	47.5	59.7	69.5	0.372	0.115	1483	2094	178
ADL-Rundle-1	MDP_Subcnn	33.4	49.9	72.4	0.344	0	2899	3230	70
	DMT	26	42.2	70.1	0.219	0.281	1697	5146	47
	Sort	20.3	31.5	72.5	0.188	0.375	1493	5812	108
	PFS(our)	39.8	38.6	73.8	0.25	0.094	1444	3998	164
ADL-Rundle-3	MDP_Subcnn	44.9	51.6	79.6	0.205	0.159	793	4752	56
	DMT	43.3	45.4	75.4	0.295	0.136	1168	4517	84
	Sort	37.4	43.4	77	0.295	0.182	1498	4765	99
	PFS(our)	34.8	34.1	80	0.273	0.159	2012	4422	197
KITTI-16	MDP_Subcnn	50	66.6	70.3	0.353	0.059	262	566	22
	DMT	44.7	60.5	69.3	0.235	0	232	690	19
	Sort	34.6	42.8	70.1	0.118	0.059	144	938	30
	PFS(our)	50	56.7	73.2	0.235	0.059	268	544	39

Table 4. *Cont.*

Sequence	Algorithm	MOTA	IDF1	MOTP	MT	ML	FP	FN	ID Sw
KITTI-19	MDP_Subcnn	40.9	61.8	68.1	0.242	0.081	1143	1965	51
	DMT	45.7	55.5	72.8	0.306	0.097	884	1946	72
	Sort	29.1	46.4	68.4	0	0.258	855	2852	79
	PFS(our)	37.8	46.4	69	0.274	0.129	1327	1890	109
Venice-1	MDP_Subcnn	42.6	48.4	76	0.353	0.235	729	1867	21
	DMT	32.4	38	70.9	0.294	0.412	527	2538	18
	Sort	24.7	24.4	69.7	0.118	0.471	485	2888	62
	PFS(our)	43.7	40.1	75.8	0.294	0.176	435	2071	64
TOTAL	MDP_Subcnn	47.5	55.7	74.2	0.3	0.186	8631	22,969	628
	DMT	44.5	49.2	72.9	0.347	0.221	8088	25,335	684
	Sort	33.4	40.4	72.1	0.117	0.309	7318	32,615	1001
	PFS(our)	48.1	45	74.7	0.327	0.15	8422	22,074	1372

It can be seen that the proposed algorithm has high accuracy. The actual effect of the algorithm on the MOT2015 dataset is shown in Figure 8.

Figure 8. Tracking result on 2DMOT15 dataset.

5. Conclusions

In this paper, we propose a pedestrian flow tracking and statistics algorithm for front-down monocular camera. The algorithm relies on convolutional neural network for real-time pedestrian detection, and uses Kalman filter linear motion model and data association algorithm to track pedestrian targets. Finally, a counting method based on virtual blocks is proposed to complete pedestrian flow statistics. We use real scene videos to evaluate the counting performance of the algorithm. At the same time, we compare the detection and tracking performance of the algorithm with other algorithms using a public dataset, MOT 2015, which proves the effectiveness of the algorithm. The experiment results show that the algorithm has good accuracy and real-time performance, and has high application value. Although the algorithm has achieved good results, there are still some shortcomings, which need further improvement. Since the accuracy of the algorithm will be affected when the pedestrian is seriously occluded, future work is to further improve the tracking accuracy of the algorithm in the case of serious occlusion by means of optical flow and re-identifying appearance model.

Author Contributions: Conceptualization, M.H.; Methodology, M.H.; Software, M.H.; Validation, M.H., B.H. and Z.C.; Formal Analysis, M.H.; Investigation, M.H.; Resources, B.H. and Z.C.; Data Curation, M.H.; Writing–Original Draft Preparation, M.H.; Writing–Review and Editing, H.L.; Visualization, M.H.; Supervision, H.L.; Project Administration, H.L.; Funding Acquisition, H.L.

Funding: This research was funded by Hainan Heaven Reward Security Technology Co. Ltd.

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

References

1. Del Pizzo, L.; Foggia, P.; Greco, A.; Percannella, G.; Vento, M. Counting people by RGB or depth overhead cameras. *Pattern Recognit. Lett.* **2016**, *81*, 41–50. [CrossRef]
2. Coşkun, A.; Kara, A.; Parlaktuna, M.; Ozkan, M.; Parlaktuna, O. People counting system by using kinect sensor. In Proceedings of the 2015 IEEE International Symposium on Innovations in Intelligent SysTems and Applications (INISTA), Taipei, Taiwan, 2–4 September 2015; pp. 1–7.
3. Verma, N.K.; Dev, R.; Maurya, S.; Dhar, N.K.; Agrawal, P. People Counting with Overhead Camera Using Fuzzy-Based Detector. In *Computational Intelligence: Theories, Applications and Future Directions—Volume I*; Springer: Berlin, Germany, 2019; pp. 589–601.
4. Kopaczewski, K.; Szczodrak, M.; Czyzewski, A.; Krawczyk, H. A method for counting people attending large public events. *Multimed. Tools Appl.* **2015**, *74*, 4289–4301. [CrossRef]
5. Beymer, D. Person counting using stereo. In Proceedings of the Workshop on Human Motion, Austin, TX, USA, 7–8 December 2000; pp. 127–133.
6. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proc. IEEE* **1998**, *86*, 2278–2324. [CrossRef]
7. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. Imagenet classification with deep convolutional neural networks. In Proceedings of the Advances in Neural Information Processing Systems, Lake Tahoe, NV, USA, 3–6 December 2012; pp. 1097–1105.
8. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
9. Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; Rabinovich, A. Going deeper with convolutions. In Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 1–9.
10. 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.
11. Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K.Q. Densely connected convolutional networks. In Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Honolulu, HI, USA, 21–26 July 2017, pp. 2261–2269.
12. Bochinski, E.; Eiselein, V.; Sikora, T. High-speed tracking-by-detection without using image information. In Proceedings of the 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Lecce, Italy, 29 August–1 September 2017; pp. 1–6.
13. Stauffer, C.; Grimson, W.E.L. Adaptive background mixture models for real-time tracking. In Proceedings of the 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), Fort Collins, CO, USA, 23–25 June 1999; p. 2246.
14. Barnich, O.; Van Droogenbroeck, M. ViBe: A powerful random technique to estimate the background in video sequences. In Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing ICASSP 2009, Taipei, Taiwan, 19–24 April 2009; pp. 945–948.
15. Dalal, N.; Triggs, B. Histograms of oriented gradients for human detection. In Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR 2005, San Diego, CA, USA, 20–25 June 2005; Volume 1, pp. 886–893.
16. Zhou, C.; Yuan, J. Bi-box Regression for Pedestrian Detection and Occlusion Estimation. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 135–151.
17. 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.* **2017**, *6*, 1137–1149. [CrossRef] [PubMed]

18. Dai, J.; Li, Y.; He, K.; Sun, J. R-fcn: Object detection via region-based fully convolutional networks. In Proceedings of the Advances in Neural Information Processing Systems, Barcelona, Spain, 5–10 December 2016; pp. 379–387.
19. Lin, T.Y.; Dollár, P.; Girshick, R.; He, K.; Hariharan, B.; Belongie, S. Feature pyramid networks for object detection. In Proceedings of the CVPR, Honolulu, HI, USA, 21–26 July 2017; Volume 1, p. 4.
20. 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, 8–16 October 2016; Springer: Berlin, Germany, 2016; pp. 21–37.
21. Fu, C.Y.; Liu, W.; Ranga, A.; Tyagi, A.; Berg, A.C. DSSD: Deconvolutional single shot detector. *arXiv* **2017**, arXiv:1701.06659.
22. Li, Z.; Zhou, F. FSSD: Feature Fusion Single Shot Multibox Detector. *arXiv* **2017**, arXiv:1712.00960.
23. Hu, W.; Li, X.; Luo, W.; Zhang, X.; Maybank, S.; Zhang, Z. Single and multiple object tracking using log-Euclidean Riemannian subspace and block-division appearance model. *IEEE Trans. Pattern Anal. Mach. Intell.* **2012**, *34*, 2420–2440. [PubMed]
24. Zhang, L.; van der Maaten, L. Structure preserving object tracking. In Proceedings of the IEEE Conference On Computer Vision And Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 1838–1845.
25. Berclaz, J.; Fleuret, F.; Turetken, E.; Fua, P. Multiple object tracking using k-shortest paths optimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **2011**, *33*, 1806–1819. [CrossRef] [PubMed]
26. Breitenstein, M.D.; Reichlin, F.; Leibe, B.; Koller-Meier, E.; Van Gool, L. Robust tracking-by-detection using a detector confidence particle filter. In Proceedings of the 2009 IEEE 12th International Conference on Computer Vision, Kyoto, Japan, 29 September–2 October 2009; pp. 1515–1522.
27. Ess, A.; Leibe, B.; Schindler, K.; Van Gool, L. Robust multiperson tracking from a mobile platform. *IEEE Trans. Pattern Anal. Mach. Intell.* **2009**, *31*, 1831–1846. [CrossRef] [PubMed]
28. Choi, W.; Pantofaru, C.; Savarese, S. A general framework for tracking multiple people from a moving camera. *IEEE Trans. Pattern Anal. Mach. Intell.* **2013**, *35*, 1577–1591. [CrossRef] [PubMed]
29. Khan, Z.; Balch, T.; Dellaert, F. MCMC-based particle filtering for tracking a variable number of interacting targets. *IEEE Trans. Pattern Anal. Mach. Intell.* **2005**, *27*, 1805–1819. [CrossRef] [PubMed]
30. Kuo, C.H.; Huang, C.; Nevatia, R. Multi-target tracking by on-line learned discriminative appearance models. In Proceedings of the 2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), San Francisco, CA, USA, 13–18 June 2010; pp. 685–692.
31. Milan, A.; Roth, S.; Schindler, K. Continuous energy minimization for multitarget tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2014**, *36*, 58–72. [CrossRef] [PubMed]
32. Ullah, M.; Cheikh, F.A.; Imran, A.S. Hog based real-time multi-target tracking in bayesian framework. In Proceedings of the 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), Colorado Springs, CO, USA, 23–26 August 2016; pp. 416–422.
33. Bae, S.H.; Yoon, K.J. Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 595–610. [CrossRef] [PubMed]
34. Kryjak, T.; Komorkiewicz, M.; Gorgon, M. Hardware-software implementation of vehicle detection and counting using virtual detection lines. In Proceedings of the 2014 Conference on IEEE Design and Architectures for Signal and Image Processing (DASIP), Madrid, Spain, 8–10 October 2014; pp. 1–8.
35. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. *arXiv* **2018**, arXiv:1804.02767.
36. He, Y.; Zhang, X.; Sun, J. Channel pruning for accelerating very deep neural networks. In Proceedings of the International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; Volume 2.
37. Bodla, N.; Singh, B.; Chellappa, R.; Davis, L.S. Soft-nms—Improving object detection with one line of code. In Proceedings of the 2017 IEEE International Conference on Computer Vision (ICCV), Venice, Italy, 22–29 October 2017; pp. 5562–5570.
38. Leal-Taixé, L.; Milan, A.; Reid, I.; Roth, S.; Schindler, K. Motchallenge 2015: Towards a benchmark for multi-target tracking. *arXiv* **2015**, arXiv:1504.01942.
39. Xiang, Y.; Alahi, A.; Savarese, S. Learning to track: Online multi-object tracking by decision making. In Proceedings of the IEEE International Conference On Computer Vision, RegióN Metropolitana, Chile, 11–18 December 2015; pp. 4705–4713.

40. Kim, H.U.; Kim, C.S. CDT: Cooperative detection and tracking for tracing multiple objects in video sequences. In Proceedings of the European Conference on Computer Vision, Amsterdam, The Netherlands, 8–16 October 2016; Springer: Berlin, Germany, 2016; pp. 851–867.
41. Bewley, A.; Ge, Z.; Ott, L.; Ramos, F.; Upcroft, B. Simple online and realtime tracking. In Proceedings of the 2016 IEEE International Conference on. IEEE Image Processing (ICIP), Phoenix, AZ, USA, 25–28 September 2016; pp. 3464–3468.