Vision Based Multi-pedestrian Tracking Using Adaptive Detection and Clustering

Zhibo Yang and Bo Yuan

Intelligent Computing Lab., Division of Informatics
Graduate School at Shenzhen, Tsinghua University
Shenzhen 518055, P.R. China
yangzhibo450@gmail.com, yuanb@sz.tsinghua.edu.cn

Abstract. This paper proposes a novel vision based multi-pedestrian tracking scheme in crowded scenes, which are very common in real-world applications. The major challenge of the multi-pedestrian tracking problem comes from complicated occlusions, cluttered or even changing background. We address these issues by creatively combining state-of-the-art pedestrian detectors and clustering algorithms. The core idea of our method lies in the integration of local information provided by pedestrian detector and global evidence produced by cluster analysis. A prediction algorithm is proposed to return the possible locations of missed target in offline detection, which will be re-detected by online detectors. The pedestrian detector in use is an online adaptive detector mainly based on texture features, which can be replaced by more advanced ones if necessary. The effectiveness of the proposed tracking scheme is validated on a real-world scenario and shows satisfactory performance.

Keywords: Pedestrian Tracking, Detection, Crowded Scene, Clustering.

1 Introduction

In recent years, vision based multi-pedestrian tracking has received growing interests from the computer vision community. Its typical applications include railway transport security, pedestrian traffic management, detection of overcrowded situations and people number counting. However, multi-pedestrian tracking in complicated environments still remains as a challenging task due to factors such as frame loss, occlusions, unusual appearances etc.

Significant efforts have been devoted to the detection and tracking of individuals in groups [1-3]. In general, the number of objects to be identified is small and the shape segmentation methods in use cannot separate individuals well in very crowded scenes (e.g., Fig. 1). It is also possible to only employ the tracking module, which may be based on sampling and particle filtering [4], MCMC [5] or greedy dynamic programming [6]. The tracking performance depends on the initial state provided by other detectors and none of these methods can guarantee reliable performance on overlapping targets. Furthermore, except the study based on linear programming [7], these approaches often perform badly when targets appear discontinuously in the video.

H. Yin et al. (Eds.): IDEAL 2013, LNCS 8206, pp. 58–66, 2013.

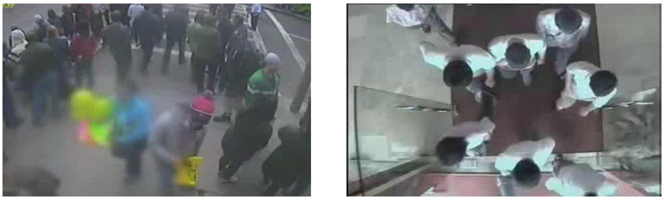

Fig. 1. Examples of crowded scenes (left: cross road; right: company main entrance)

To handle occlusions, Sidla et al. [8] used active shape models and the KLT (Kanade-Lucas-Tomasi) tracking algorithm to compute regions of interest and linked new shape locations to their corresponding trajectories. Lin et al. [9] estimated the number of people in crowded scenes using perspective transforms. Their approaches are designed for counting purpose and cannot track pedestrians in real time. With the multi-camera approach [10, 11], overlapping people can be tracked by different cameras and individual trajectories are processed separately over long sequences. In this paper, the objective is to realize multiple tracking with a single camera.

Inspired by the idea of data association [12] and adaptive detection [13], we decompose multiple objects tracking into three steps. In the first step, a detector trained in an offline manner estimates the number and locations of targets. Pedestrian detection has been extensively studied and a range of ready-to-use detectors are available. Previous studies have shown clearly that tracking algorithms based on whole body detection using shape features [8] are limited in their performance as soon as pedestrians are partially occluded. We adopt the idea proposed in [14] but use Haar-like features to train our two-part body detectors to solve the issue of partial occlusion. The second step considers the most common problem in practice where the detector produces false-negative results. A target validation module is used to predict the possible location of the missing target and redetect the target in the possible positions by template matching. In the third step, target locations and time information are combined to form trajectories using sequential leader clustering (SLC). The outliers (false-positive results) can be easily detected by this method.

The rest of the paper is organized as follows: Section 2 introduces the framework of our approach and presents some basic modules such as AdaBoost for training the pedestrian detector and sequential leader clustering. The key algorithms are discussed in Section 3 and 4. The experiment results will present in Section 5 and this paper is concluded in Section 6 with some discussion and a list of directions for future work.

2 Framework

The general framework of our approach is shown in Fig. 2. We use a state-of-the-art pedestrian detection method [14] based on body parts representation. The detection module consists of two stages: detection of body parts and combination of body parts. The major advantage of this approach is that it can deal with partial occlusions (e.g., when the lower part of an individual is occluded, the upper part may still be detected). The human body is divided into two parts: head-shoulder (Ω-like part) and leg (Π-like part). Haar-like features are extracted from the training set containing both pedestrians and non-pedestrians samples and two classifiers are trained to classify these two parts based on AdaBoost, which selects a small number of critical Haar-like features including edge features, line features and center-surround features.

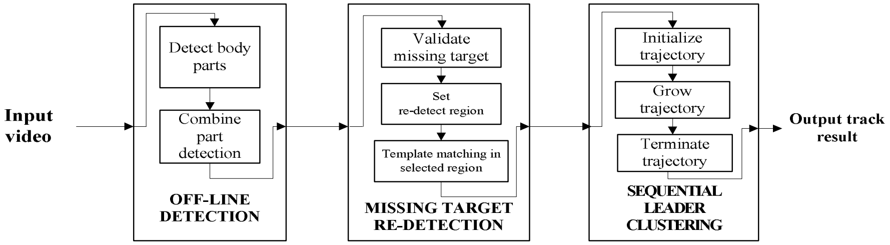

Fig. 2. A schematic framework of the proposed pedestrian tracking system

Regardless of the specific detector in use, the missing target validation is indispensible. It is very common that an individual may fail to be detected in a certain frame but is recognizable in previous and following ones. These false negative results are often isolated in time and should be re-detected (the second step in Fig. 2). How to confirm whether a false negative detection (a target is missing) or a false positive detection exists is an interesting issue. We model the belonging likelihood of new observations based on spatial, motion and appearance consistencies. Once new detections violate the constraints constructed by the likelihood model, the tracking system will indicate the missing targets and set the search area predicted by our model.

Note that the detector itself does not distinguish pedestrians and we propose to use clustering such as SLC to distinguish people from each other. Each pedestrian at each time step is specified by a data point and clustering is used to form a continuous trajectory for each pedestrian. To the best of our knowledge, this is the first time that SLC is employed to solve complex tracking problems.

3 Missing Target Re-detection

3.1 Validation of Detection Results

For a fixed position camera, a target's movement in the monitor region has to obey certain rules. Moving objects can only enter or leave the screen from the edge line rather than suddenly appear at the center of the screen. In general, a pedestrian walks at a speed of 5-6 km/h. The appearance especially the size of a target changes little within limited time frames and we aim to get a sequence of rectangular boxes for each detection result. However, a detector is usually not sufficient for this purpose and a validation module is required to find the false detections that violate the above rules. The observed state P_t^i of the i^{th} detected pedestrian at time t is defined as:

$$P_t^i = \{c_t^i, s_t^i, v_t^i\}, \qquad c = (x, y), \quad s = (w, h), \quad v = (dx, dy) \tag{1}$$

Where c is the coordinate of the rectangle's center, s is the width and the height of the rectangle box and v is the velocity of the moving rectangle. To validate the false negative detection and identify overlapping situations, new detections are compared with historic detections using similarity measurement. Three distance functions are proposed to measure the similarity between two rectangles (Eq. 2). D_1: *spatial proximity*. The spatial proximity describes the consecutive movement of the same person, measured by the Euclidean distance between locations. D_2: *box size*. Different pedestrians are unlikely to have identical box size and D_2 is defined as the normalized difference between pedestrian sizes. D_3: *velocity coherence*. The velocity vectors contain the direction information that is important for dealing with occlusions.

$$D_1 = \|c_m - c_n\|, \qquad D_2 = \|s_m - s_n\|, \qquad D_3 = \|v_m - v_n\| \tag{2}$$

Missing targets: when new targets are detected at time t, historic detections have been labeled using SLC. After comparing all m new rectangles with the last few rectangles in the trajectory of the i^{th} pedestrian, if the condition specified by Eq. 3 is met (T is threshold), it is assumed that the target corresponding to the i^{th} pedestrian is missing and re-detection is required in its neighborhood.

$$D_1^{i,j} > T_1, and \quad D_2^{i,j} > T_2, and \quad D_3^{i,j} > T_3, where \quad j = (0 \cdots m) \tag{3}$$

Full occlusions: the employed detector can solve most partially occlusions and we only need to focus on totally occlusions. In the simplest situations, the i^{th} pedestrian stands between the j^{th} pedestrian and the camera. In this case, there may be a detection n that satisfies the conditions in Eq. 4 and a flag will be set so that the subsequent clustering module (Section 4) can correctly handle this occlusion. It is easy to extend the two pedestrian occlusions to multi-pedestrian occlusions, and we duplicate the observation the times same to the number of being occluded pedestrians.

$$D_1^{k,m} < T_1, and \quad D_2^{k,m} < T_2, and \quad D_3^{k,m} < T_3, where \quad k = (i, j \cdots) \tag{4}$$

3.2 Re-detection

Once missing pedestrians are identified by the method specified in Section 3.1, a search process is carried out within a limited region based on the spatial and velocity information instead of the entire image to reduce the computational cost. Suppose that the missing target's last state is P_t, center at (x_t, y_t), size (w_t, h_t) and the average velocity of this target's movement in one step is (*dx, dy*). The most likely center of the missing target is set to (x_t+*dx*, y_t+*dy*). An empirical re-detection region is described by Eq. 5.

$$region:\{x_t + dx \pm 3w_t,\ y_t + dy \pm 3h_t\} \tag{5}$$

Next, the last two detections of the pedestrian are stored in the form of color histogram and a template matching method [15] is used to traverse the re-detection region to identify the missing target.

4 SLC for Multiple Tracking

Fig. 3(left) shows a scenario where the camera is equipped on top of the main entrance of an office building. In Fig. 3(middle), the x-axis and y-axis represent the location of pedestrian and the vertical axis represents the time. A trajectory is created by sequentially adding pedestrian records into the cluster corresponding to the same pedestrian. Fig. 3(right) shows an example where there are three trajectories (clusters) represented by red, blue and green respectively and a new record (black cross) to be identified. In this paper, we use SLC as the clustering technique, which measures the similarity between the new record and each of the three clusters and assign it to the most similar cluster (trajectory).

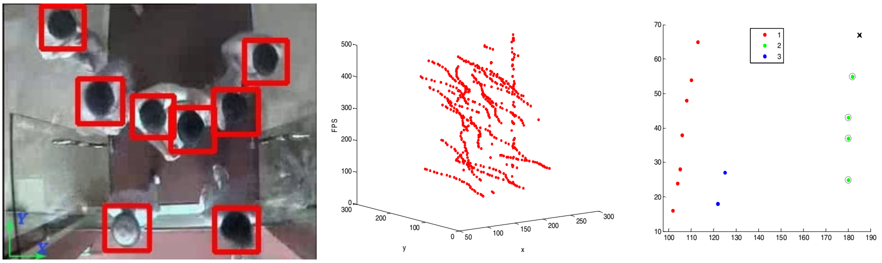

Fig. 3. A real-world scenario where the camera is equipped on top of the entrance (left), the 3D representation of pedestrian records (middle) and trajectory expansion (right)

SLC is used to cluster the detections into a number of clusters in an incremental manner, corresponding to the trajectories of unique pedestrians (Algorithm 1). Some modification is made to the original SLC so that whenever a full occlusion is detected (Eq.4, Section 3.1), duplicates of the original detection will be created with each one assigned to one of the intersecting trajectories.

Algorithm 1. SLC for Multiple Tracking

dist(x,y)—function for measuring similarity between objects x and y
createCluster—function for creating a cluster that includes one element x
addToCluster—procedure for adding element x to cluster c
selectFrom— procedure that selects an element from clusters
e—threshold of closeness of exemplar
n—the total detections in time t
bigNum—the threshold for creating a new cluster
cm—cluster with mark bit c_flag in the end
c_flag—a flag for indicating a cluster cm has receive a new element

```
SLCinTracking(x[i], clusters, e)
for (i=0; i<n; i++)
{
    If empty(clusters) then
            clusters = {createCluster(x[i])};
         return;
    dmin=bigNum;
       cm=selectFrom(clusters);
    for c∈ clusters &c_flag≠1 do
        dd=dist(x[i],c);
        ifdmin>dd&dd≦e then
        dmin=dist(x,c); cm=c;
    if dmin<bigNum then
        addToCluster(x[i],cm); c_flag=1;
    else
    createCluster(x[i]);
}
```

5 Experiment

5.1 Standard Benchmark Results

In this section, we present the results of our method on the CAVIAR dataset [16]. This dataset is very challenging due to sever occlusions and cluttered and complicated backgrounds. The CAVIAR dataset contains 26 video sequences of a walkway in a shopping center taken by a single fixed camera with frame size 385×288 and frame rate 25fps. For comparison purpose, the results from some previous studies [12, 14] are also included. We evaluated the tracking performance from three aspects: completely tracked pedestrians (CT), completely lost pedestrians (CL), and partially tracked pedestrians (PT). The experiment results are shown in Table 1, where GT represents the ground truth.

Table 1. Experiment results on CAVIAR

Method	GT	CT	PT	CL	CT RATE
Wu et al. [14]	140	106	25	9	75.17%
Zhang et al. [12]	140	104	29	7	74.28%
Our Method	140	113	19	8	80.72%

We can see that our method achieved a higher CT rate, capable of fully tracking more pedestrians than existing methods. This is mainly due to introduction of the re-detection mechanism. Fig. 4 shows some of the recognition examples.

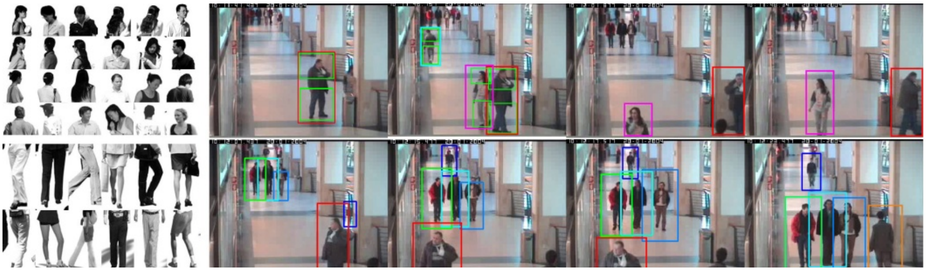

Fig. 4. Examples of the tracking results on the CAVIAR dataset

5.2 Surveillance Applications

We applied the proposed method to a real-world application on pedestrian counting based on the video from the CCTV monitoring a large company's main entrance. Compared to the CAVIAR dataset, the back ground was much cleaner, with less number of occlusion situations due to position of the camera but the people were more crowded in the video (Fig. 5). The frame size was 320×240 and frame rate was 15fps. A large set of positive and negative samples were collected for training the detector. The overall precision was more than 96% and the processing time for each frame was around 50ms on a 2G Hz dual core computer with 2G memory.

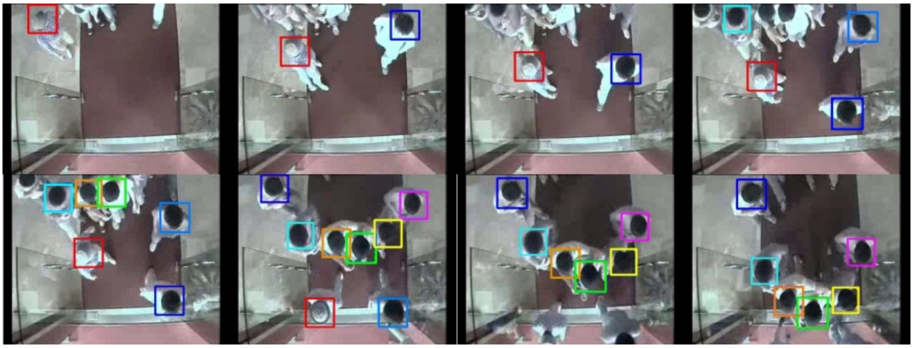

Fig. 5. Examples of the pedestrian counting application

6 Conclusions

In this paper, we applied the sequential leader clustering method to multiple pedestrian tracking, which provides a new direction for the research in this area and is much simpler than other existing techniques. The proposed method can also validate the detection results and, if necessary, re-detect the missing targets at a very low cost. The experiment results on a standard benchmark video dataset and a real-world application show that the precision of our method is satisfactory in most scenes with fast processing speed. In the future, we will focus on investigating more advanced pedestrian detectors, incorporating more cues in validation and testing other clustering or classifying methods to further improve the tracking performance.

Acknowledgement. This work was partially conducted while the first author was working at the R&D department of MICROPROFIT Company. We are also grateful to Mr. Haoran Xin for his support during the test of the prototype system.

References

1. Zhao, T., Nevatia, R.: Bayesian Human Segmentation in Crowded Situations. In: 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 459–466 (2003)
2. Rittscher, J., Tu, P.H., Krahnstoever, N.: Simultaneous Estimation of Segmentation and Shape. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 486–493 (2005)
3. Leibe, B., Seemann, E., Schiele, B.: Pedestrian Detection in Crowded Scenes. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 878–885 (2005)
4. Chang, C., Ansari, R., Khokhar, A.: Multiple Objects Tracking with Kernel Particle Filter. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 566–573 (2005)
5. Ge, W., Collins, R.T.: Multi-Target Data Association by Tracklets with Unsupervised Parameter Estimation. In: 19th British Machine Vision Conference, pp. 93.1–93.10 (2008)
6. Shafique, K., Shah, M.: A Noniterative Greedy Algorithm for Multiframe Point Correspondence. IEEE Transactions on Pattern Analysis and Machine Intelligence 27(1), 51–65 (2005)
7. Jiang, H., Fels, F., Little, J.J.: A Linear Programming Approach for Multiple Object Tracking. In: 2007 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2007)
8. Sidla, O., Lypetskyy, Y., Brandle, N., Seer, S.: Pedestrian Detection and Tracking for Counting Applications in Crowded Situations. In: 2006 IEEE International Conference on Video and Signal Based Surveillance, p. 70 (2006)
9. Lin, S.F., Chen, J.Y., Chao, H.X.: Estimation of Number of People in Crowded Scenes Using Perspective Transformation. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 31(6), 645–654 (2001)
10. Fleuret, F., Berclaz, J., Lengagne, R., Fua, P.: Multicamera People Tracking with A Probabilistic Occupancy Map. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2), 267–282 (2008)
11. Berclaz, J., Fleuret, F., Turetken, E., Fua, P.: Multiple Object Tracking Using K-shortest Paths Optimization. IEEE Transactions on Pattern Analysis and Machine Intelligence 33(9), 1806–1819 (2011)

12. Zhang, L., Li, Y., Nevatia, R.: Global Data Association for Multi-object Tracking Using Network flows. In: 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
13. Kalal, Z., Mikolajczyk, K., Matas, J.: Face-tld: Tracking-Learning-Detection Applied to Faces. In: 2010 IEEE International Conference on Image Processing, pp. 3789–3792 (2010)
14. Wu, B., Nevatia, R.: Detection and Tracking of Multiple, Partially Occluded Humans by Bayesian Combination of Edgelet Based Part Detectors. International Journal of Computer Vision 75(2), 247–266 (2007)
15. Pereira, S., Pun, T.: Robust Template Matching for Affine Resistant Image Watermarks. In: 2000 IEEE International Conference on Image Processing, pp. 1123–1129 (2000)
16. CAVIAR Dataset, `http://homepages.inf.ed.ac.uk/rbf/CAVIARDATA1`