Robust Person Tracking Algorithm Based on Convolutional Neural Network for Indoor Video Surveillance Systems

Rykhard Bohush(✉) and Iryna Zakharava

Polotsk State University, Blokhina Street, 29, Novopolotsk, Republic of Belarus
bogushr@mail.ru, ira99920ll@yandex.ru

Abstract. In this paper, we present an algorithm for multi person tracking in indoor surveillance systems based on tracking-by-detection approach. Convolutional Neural Networks (CNNs) for detection and tracking both are used. CNN Yolov3 has been utilized as detector. Person features extraction is performed based on modified CNN ResNet. Proposed architecture includes 29 convolutional and one fully connected layer. Hungarian algorithm is applied for objects association. After that object visibility in the frame is determined based on CNN and color features. For algorithm evaluation prepared videos that was labeled and tested using MOT evaluation metric. The proposed algorithm efficiency is illustrated and confirmed by our experimental results.

Keywords: Person · Tracking · Indoor · CNN · CUDA

1 Introduction

Object tracking task in indoor environments is far away from solving. Tracking algorithms have a variety of technical applications such as surveillance, autonomous driving systems, action recognition and etc. These practical applications claim robust object segmentation, identification and real-time processing. Multi person tracking is more challenging because of complicated background, multiple occlusions, fast object movements and dynamic background changing. Also in indoor environments there are limitations in camera position. It leads to object occlusions, person proportions disfeaturing and may complicate tracking.

MOT (Multi object Tracking) challenge analysis shows that the most common algorithm for tracking is tracking-by-detection approach [1]. MOT proposes evaluation metric for tracking algorithms and publishes evaluation results for them. A tracking-by-detection approaches use an ensemble of a detector and an assignment problem solver. For object detection can be used Haar cascades, Support Vector Machine (SVM) and CNNs. For assignment problem solving can be used object association based on "strong" and "weak" object features comparison. As "weak" features we denote the temporal shift, color and shape. As "strong" features can be used HOG, SIFT, SURF and CNN features. CNN approaches are stable to light variance, dynamic background changing and in case partial occlusions, but claim a lot of computational cost.

S. V. Ablameyko et al. (Eds.): PRIP 2019, CCIS 1055, pp. 289–300, 2019.
https://doi.org/10.1007/978-3-030-35430-5_24

2 Related Works

A tracking-by-detection approaches use a variety of techniques to solve detection and assignment. That is the reason why in this section not only algorithms dedicated to tracking but and approaches which are trying to solve the assignment problem, or person re-identification for human tracking are reviewed.

In [2], authors have proposed an algorithm, which uses MOG2 background extractor as person detector and Kalman filtering for object movement prediction. For tracking was utilized Lukas-Kannade optical flow. This approach cannot be applied for difficult indoor video due to errors in case multiple objects and dynamic background changing. Another one algorithm [3] was used cascade head detection. After that, the algorithm recalculates a human body location using stable percentage ratio. For assignment problem, solving was exploited bag of features approach. Presented approach performs better assignment than in [2] but it has limitation of the person positions that can be recognized by using only head detector. CNN based algorithms perform robust detection and assignment in case complicated articulation, light changing and etc. In [4] authors presented CNN architecture for re-identification SiameseNet which uses two identical architectures that share features due image processing. As input, this architecture has two images and as the output CNN performs binary classification two objects on these images contain one identity or not. Zhao et al. [5] use CNN PartNet that extracts features from the most distinguished part from an image. This approach for person re-identification performs good results but it leads to big computational cost. In application to tracking problem in [6] was tested SiameseNet and approach based on Faster CNN for feature extraction. Euclidean distance feature matching performs better results than SiameseNet even without training CNN for person re-identification task. In [7] authors perform object tracking using points based on the human body joints extraction using CNN. Distance between these points on two frames used for re-identification. More robust tracking performed in [8] that builds a graph consisted of spatial edges connecting the detections within a frame and temporal edges connecting detections of the same joint type over frames The body keypoints based algorithm shows more stable approach in case partial occlusions but these approaches useless in case low resolution and need a lot of computational cost.

In [9] proposed enhanced SORT [10] model which is called DeepSort. This approach is based on custom CNN architecture for “strong” feature extraction. Kalman filtering was used to predict object movements. DeepSort performs high-speed processing at 40fps but this value calculated only for tracking part. In [11] DeepSort approach with Yolov3 detector [12] and whole algorithm runs on 11.5 fps using NVIDIA GTX 1060. DeepSort shows the best result in MOT16 [13]. However, for the feature extraction it is better to use CNN architecture with residual connections that performs better results in deep architectures than fully convolutional models and helps to avoid overfitting during training [14].

We propose the person tracking algorithm for indoor video surveillance using CNNs. As person detector we decided to use Yolov3 that is has low percentage ratio of errors results and fast enough for real-time processing. The tracking was performed as an assignment problem solving and as main parameter was used similarity value.

This value contains "strong" and "weak" features. Modified ResNet34 architecture for "strong" feature extraction is used. In addition, we calculate "weak" features like the temporal shift, object width and height variation. To reduce computational time all CNN operations was performed on GPU with CUDA support.

3 Algorithm Description

The algorithm includes in four sub-tasks: object detection, assignment problem solving, tracklets update, tracklets post processing. The general scheme of our algorithm is presented in Fig. 1.

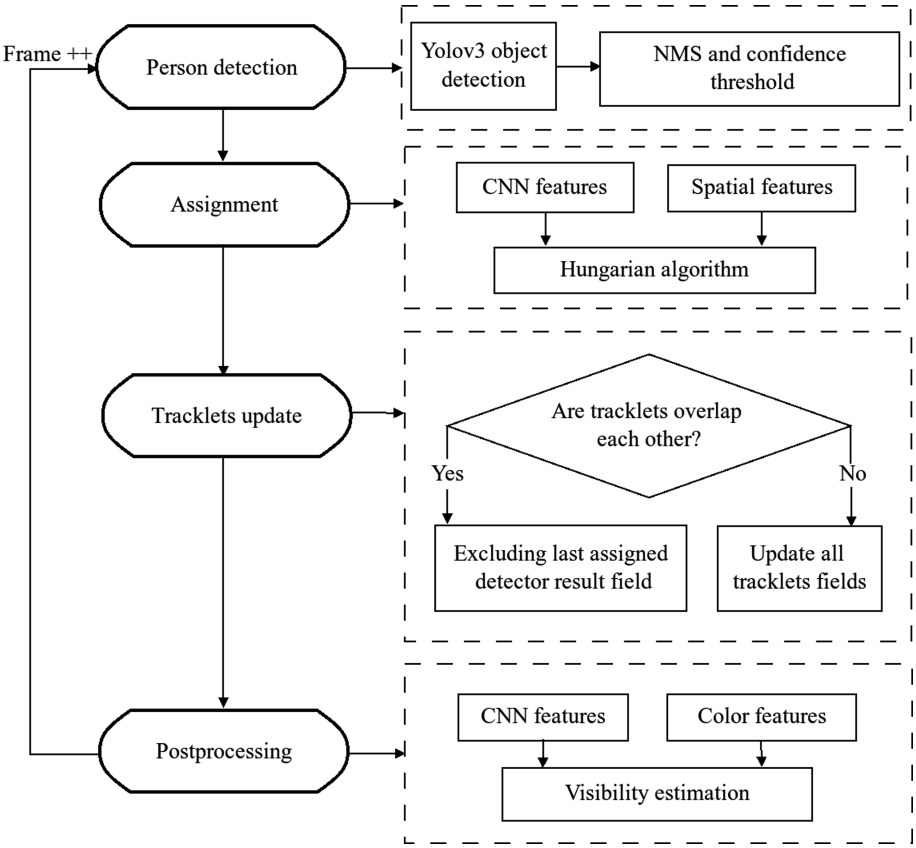

Fig. 1. The proposed algorithm flow-chart

Frames from the stationary video surveillance camera are input to the algorithm. First, people are detected in the frame. Then detected areas filtered out using Non-Maxima suppression and confidence threshold before assignment. For each pair of detections and trackers calculated similarity value that used as input for assignment

algorithm decision making. According to the association results values in the tracklets fields update with the criterion for last detector result field. The last step is used to decide on the visibility of the object in the frame and tracking results correction.

Person detection task can be presented as function of the detector from an image as:

$$\{cl, l, p\} = Dt(im), \tag{1}$$

where Dt – detection approach; cl, l, p – detector output results as cl – object class, l – object location, p – confidence score.

YOLO v3 model was used for object detection. It has fully convolutional architecture for feature extraction Darknet-53 which is containing 53 convolutional layers, 23 residual connections and achieve 93.8% top-5 accuracy. For detector was chosen main parameters: input layer size [480 × 480], confidence threshold = 80%, NMS threshold = 0.7.

For assignment problem solving of person we store information about every identity in tracklet, which includes total information about individual object. Tracklet contains the name or current tracklet index, the image of last assigned detector result, the current object image, object coordinates on current and previous frame and the visibility value and is presented as:

$$tr = \{n, im_{_{dt}}, im_{curr}, coor_{curr}, coor_{prev}, v\}, \tag{2}$$

where n – tracklet name; im_{dt} – image of last assigned detector result; im_{curr} –current object image; $corr_{curr}$ – object coordinates on current frame; $corr_{prev}$ – object coordinates on previous frame; v – visibility value.

Because our task is the indoor surveillance systems, the tracklet period of life is very important parameter. On the street surveillance systems object normally has one entry point and one exit point. In the indoor environments, this rule does not work and objects can have multiple input and output points. That is why in the indoor surveillance systems required longer period of life value for tracklet.

Tracklet life span increasing leads to some errors. Main of them is false positive results and wrong assignment to new object. To solve the first problem we calculate the visibility value using a last assigned detector result. The second one is solved by using the threshold values due to similarity matrix calculation. In addition, to prevent indexing errors after multiple occlusions image of last assigned detector result field in tracklets updates only when objects on current frame do not overlap each other.

The CNN architecture for feature extraction is presented in Fig. 2. This architecture based on ResNet-34 and constructed for person re-identification task solving. The input convolution layer [7 × 7] was replaced, because minimal sized convolutional layers performs better results in re-identification task [15]. The number of fully connected layer outputs was reduced from 1000 to 128 to transform feature map from the last convolutional layer to feature vector that describes input person image. Convolutional layers number was reduced to 29. This value was chosen during tests as minimal layers size with maximum accuracy.

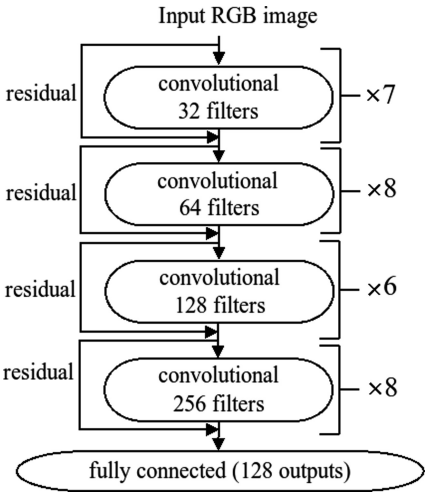

Fig. 2. The CNN architecture for feature extraction

Assignment is calculated based on similarity value as:

$$D = D_{cnn} + dx + dy + dw + dh, \tag{3}$$

where D_{cnn} – Euclidean distance between current and last aligned object CNN features; *dx, dy, dw, dh* – «weak» features – horizontal shift, vertical shift, width variance, height variance relatively.

A similarity matrix is formed after calculating all assignment values:

$$DD = \begin{pmatrix} D_{00} & \dots & D_{0K} \\ \vdots & \ddots & \vdots \\ D_{L0} & \dots & D_{LK} \end{pmatrix}, \tag{4}$$

where D_{KL} – similarity value, K – tracklets number, L – current frame person detector results number.

At the next step the association is based on Hungarian algorithm [16]:

$$a = \arg\min(tr_i, dt_{0..L}), \tag{5}$$

where t_i - tracklet; $dt_{0..L}$ – person detector results; a – assignment decision from (−1) to L. If a = (−1), then current tracklet not associated.

For robust assignment last aligned object image field in tracklet container updates only when objects do not overlap each other. To measure the overlapping value we calculate *IoU* (Intersection Over Union) as:

$$IoU = \frac{T_i \cap T_{i+1}}{T_i \cup T_{i+1}}, \tag{6}$$

where T_i and T_{i+1} are tracklet coordinates.

For visibility value "strong" and "weak" features calculation is performed. As "strong" features we use CNN features between the last aligned detector result and the object on current frame, as "weak" features was used H channel from HSV color space.

The similarity for histograms is computed as follows [17]:

$$if\ \left|H_i^{tr} - \overline{H^{tr}}\right| > \left|H_i^{dt} - \overline{H^{dt}}\right|, \quad R = \frac{1}{N^2}\sum_{i=0}^{N-1}\frac{(H_i^{dt} - \overline{H^{dt}})}{(H_i^{tr} - \overline{H^{tr}})}, \tag{7}$$

$$else: \qquad R = \frac{1}{N^2}\sum_{i=0}^{N-1}\frac{(H_i^{tr} - \overline{H^{tr}})}{(H_i^{dt} - \overline{H^{dt}})}.$$

where H^{tr} – current object color histograms; H^{dt} – last aligned object color histograms; N– number of histogram samples; $\overline{H^{tr}} = \frac{1}{N}\sum_{i=0}^{N-1} H_i^{tr};;\ \overline{H^{dt}} = \frac{1}{N}\sum_{i=0}^{N-1} H_i^{dt}$.

The visibility value can be represented as:

$$v = \begin{cases} 0 \ \ if \ \ D_{feat} < 0.3 \ and \ R > 0.2; \\ 1 \ \ in \ \ other \ \ cases, \end{cases} \tag{8}$$

where D_{feat} – Euclidean distance between current and last aligned object CNN features.

If the visibility value is equal to 1, then object still present in the frame, other values mean that object is invisible.

4 Training

Proposed CNN for person re-identification was trained on custom database that used samples from the PRID [18] and the ILIDS-VID [19]. The PRID contain 749 identities. They are similar appearing on a simple background with luck of texture features. The custom database was escalated by the ILIDS-VID which is has more complicated examples but contain only 300 identities. They are has complicated background and wear. The database contains 1030 identities approximately 100 images for each of them. Also, due training was performed color jittering for database escalation. Some samples from the PRID and the ILIDS-VID shown in Fig. 3.

CNN for person re-idintification was trained on PC with CPU Intel Core i5-8600, 3.6 GHz, 16 Gb RAM, two GPU Nvidia GTX 1060. Main parameters for training: learning rate = 0.001, momentum = 0.9. Testing database contains 300 identities, about 100 photos for each of them. With pretrained CNN for person re-idintification we achieve 99.9% accuracy. Mean loss function in the end of training was $7.4 \cdot 10^{-6}$.

Fig. 3. Samples from person re-identification dataset: (a) shown moving object on simple background from PRID; (b) challenging images with different pose and complicated background from ILIDS-VID

5 Experimental Results

For the algorithm quality evaluation we prepare indoor video sequences with various shooting conditions. Five video sequences obtained using a stationary video camera in indoor environments with various light conditions, number persons in the frame and resolution was used for testing (Fig. 4). Description for test videos presented in Table 1.

Table 1. Description for test videos

Video sequence	Resolution	Description
1 (Figure 4a–c)	[1920 × 1080]	3 persons on the video; Multiple input points for one identity; Multiple occlusions by background; Width and height variation during occlusions;
2 (Figure 4d–f)	[1920 × 1080]	3 persons on the video; Multiple occlusions by background; Multiple inter object overlapping;
3 (Figure 4g–i)	[1280 × 960]	2 persons on the video; Multiple occlusions by background; Width and height variation during occlusions;
4 (Figure 4j–l)	[1920 × 1080]	3 persons on the video; Multiple input points for one identity; Multiple occlusions by background; Long term multiple inter object overlapping;
5 (Figure 4m–o)	[3840 × 2160]	2 persons on the video; Multiple inter object overlapping; Light variance

Fig. 4. Some scenes from test video sequences

For the algorithm evaluation is used MOT metric from. Therefore, we labeled these videos according to MOT requirements. This metric was developed for a multi object tracking algorithms evaluation and contain the following parameters [20]:

- The ratio of correctly identified detections over the average number of ground-truth and computed detections (IDF);
- Multiple Object Tracking Accuracy. This measure combines three error sources: false positives, missed targets and identity switches (MOTA);

- Multiple Object Tracking Precision. The misalignment between the annotated and the predicted bounding boxes (MOTP);
- The total number of false positives (FP);
- The total number of false negatives (missed targets) (FN);
- Processing speed n frames per second excluding the detector (Freq).

Our algorithm was implemented in C++ using OpenCV and dlib [21]. All CNN operations were performed on GPU with CUDA support. With our implementation was collected MOT evaluation results that shown in Table 2. For the video 1 and 5 MOTA values lower than the other, because of multiple inter objects occlusions with fast shape changing during it. For the other test videos where objects shape does not significantly change during occlusions MOTA values are higher.

Table 2. MOT experimental results for each test video sequences

Number of video sequences	IDF1	FP	FN	MOTA	MOTP
1	70	173	203	83.2	78.9
2	96.7	24	51	93.5	76.8
3	98.7	12	19	97.4	81
4	92.8	8	27	97.1	82.1
5	93	88	183	88.2	77.8

Processed frames parts are given in Fig. 5. Since our approach start indexing from 0, for DeepSort algorithm the first index there is 1. Due to processing we do not have to see indices larger than 0 in cases our algorithm (Fig. 5d–f) and 1 in case DeepSort (Fig. 5a–c) for one person in video. If we will see them it will show us errors like false positive detector results or poor assignment. DeepSort algorithm makes the new objects when one person is partially hidden from background (Fig. 5b), or cannot track it (Fig. 5c). Our algorithm processed one object with the same index without false negative results (Fig. 5d–f). The second example is presented for three object tracking in video (Fig. 5g–l). Figure 5g–i demonstrate that indices are bigger than 3 for DeepSort. Also it has a lot indices switching between different identities. Figure 5j–l show that indices stay on the tracked objects even with long time occlusions for proposed algorithm.

The MOT average values show algorithm effectiveness for all test videos. In Table 3 we show the final MOT metrics for Deep Sort and the proposed algorithm.

Table 3. Comparison MOT metrics for algorithms

Algorithm	MOTA	MOTP	FP	FN	IDF	Freq
DeepSort [10]	87.2	**81.1**	**113**	997	60.1	**60**
Proposed	**90.3**	79.1	305	**403**	**87.9**	**60**

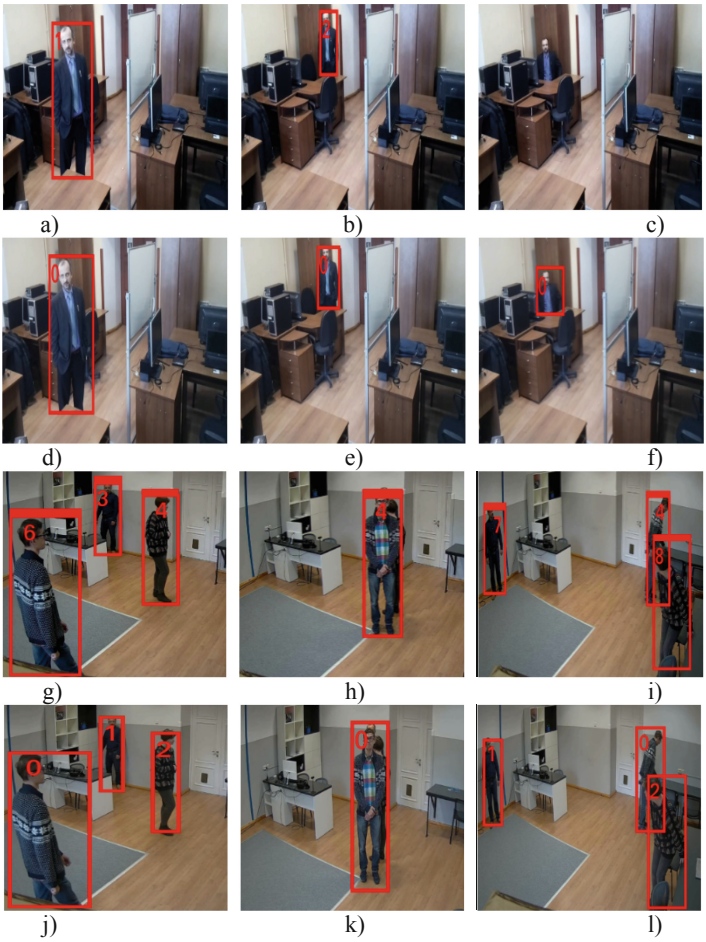

Fig. 5. Examples for person tracking: DeepSort (a,b,c,g,h,i) and proposed approach (d,e,f,j,k,l)

The data obtained (Table 3) show that our algorithm increases the value of correctly identified objects. Also the number of false negative results is reduced too. The false positive numbers is higher than in DeepSort because of higher life time for the tracklet which is help to track objects even if they are hidden behind the background or another object in small period of time. The value F_{req} is processing speed in MOT evaluation metrics detector part do not taken into account. Processing speed including detection task for our approach runs at 25 fps using NVIDIA GTX 1060.

6 Conclusion

Person tracking for indoor surveillance is challenging task. We have presented our real-time tracking-by-detection algorithm which is specified for indoor environments. Our algorithm contains following steps: person detection, assignment problem solving, tracklets formation, tracklets post processing. Object detection performed with CNN Yolov3. For assignment proposed CNN architecture based on ResNet-34 and pre-trained on person re-identification dataset. For the algorithm evaluation was prepared 5 test video that filmed in indoor environments with various light conditions, number persons in the frame and resolution. Experimental video was prepared and labeled according to MOT requirements. With our algorithm we achieve following results using MOT metric: MOTA = 90.3, MOTP = 79.1, FP = 305, FN = 403, IDF = 87.9. To reduce computational cost all CNN operations was performed on GPU with CUDA technology.

Consequently, proposed approach has perspectives in indoor surveillance systems. For the future work it is planned to apply our algorithm for multi camera tracking task.

References

1. MOTChallenge: The Multiple Object Tracking Benchmark. https://motchallenge.net. Accessed 14 Aug 2019
2. Miguel, M.D., Brunete, A., Hernando, M., Gambao, E.: Home camera-based fall detection system for the elderly. Sensors **17**(12), 2864 (2017)
3. Kuplyakov, D., Shalnov, E., Konushin, A.: Markov chain Monte Carlo based video tracking algorithm. Program. Comput. Softw. **43**(4), 224–229 (2017)
4. Tao, R., Gavves, E., Smeulders, A.W.: Siamese instance search for tracking. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1420–1429. IEEE, Las Vegas (2016)
5. Zhao, L., Li, X., Zhuang, Y., Wang, J.: Deeply-learned part-aligned representations for person re-identification. In: 2017 IEEE International Conference on Computer Vision (CVPR), pp. 3239–3248. IEEE, Venice (2017)
6. Chahyati, D., Fanany, M.I., Arymurthy, A.: Tracking people by detection using CNN features. Proc. Comput. Sci. **124**, 167–172 (2017)
7. Insafutdinov, E., et al.: ArtTrack: articulated multi-person tracking in the wild. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 1293–1301. IEEE, Honolulu (2016)
8. Iqbal, U., Milan, A., Gall, J.: PoseTrack: joint multi-person pose estimation and tracking. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4654–4663. Honolulu (2016)
9. Wojke, N., Bewley, A., Paulus, D.: Simple online and real time tracking with a deep association metric. In: IEEE International Conference on Image Processing (ICIP), pp. 3645–3649. IEEE, Beijing (2017)
10. Bewley, A., Ge, Z., Ott, L., Ramos, F.T., Upcroft, B.: Simple online and real time tracking. In: 2016 IEEE International Conference on Image Processing (ICIP), pp. 3464–3468. IEEE, Phoenix (2016)
11. Real-time Multi-person tracker using YOLO v3 and deep_sort with tensorflow. https://github.com/Qidian213/deep_sort_yolov3. Accessed 14 Aug 2019

12. YOLOv3: An Incremental Improvement. Source. https://arxiv.org/abs/1804.02767. Accessed 14 Aug 2019
13. MOT16 Results. https://motchallenge.net/results/MOT16. Accessed 14 Aug 2019
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778. IEEE, Las Vegas (2016)
15. Wu, L., Chunhua, S., Hengel, A.: PersonNet: person re-identification with deep convolutional neural networks. https://arxiv.org/pdf/1601.07255.pdf. Accessed 14 Aug 2019
16. Kuhn, H.W.: The Hungarian method for the assignment problem. Naval Res. Logistics Q. **2**, 83–97 (1995)
17. Bogush, R., Maltsev, S.: Minimax criterion of similarity for video information processing. In: Siberian Conference on Control and Communications, pp. 120–127. IEEE, Tomsk (2007)
18. Person Re-ID (PRID) Dataset. https://www.tugraz.at/institute/icg/research/team-bischof/lrs/downloads/prid11/. Accessed 14 Aug 2019
19. iLIDS Video re-IDentification (iLIDS-VID) Dataset. http://www.eecs.qmul.ac.uk/∼xiatian/downloads_qmul_iLIDS-VID_ReID_dataset.html. Accessed 14 Aug 2019
20. Keni, B., Stiefelhagen, R.: Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. **1**, 1–10 (2008)
21. King, D.W.: Dlib-ml. machine learning toolkit. J. Mach. Learn. Res. **10**, 1755–1758 (2009)