RetryTRACK: Recovering Misses in Multi-camera Pedestrian Tracking

Isabella de Andrade
Voxar Labs, Centro de Informática
Universidade Federal de Pernambuco
isfa@cin.ufpe.br

João Paulo Lima
Departamento de Computação
Universidade Federal Rural de Pernambuco
joao.mlima@ufrpe.br

Veronica Teichrieb
Voxar Labs, Centro de Informática
Universidade Federal de Pernambuco
vt@cin.ufpe.br

Abstract

Tracking pedestrians commonly relies on detection algorithms. However, these algorithms are not always correct and may miss some pedestrians. Although using multiple cameras is a way to handle this, some failures still occur. Thus, it is desirable that the tracker attempt to fix the detections. This work proposes an online and unsupervised module to recover missing detections during tracking. The module applies linear extrapolation and Gaussian process regression techniques to produce new smoothed coordinates. We attached the module to a multi-camera baseline tracker and evaluated it on the WILDTRACK dataset. The multiple object tracking accuracy was improved by 2.42% with the addition of the module. Besides, this strategy recovered 20.3% of missing detections, demonstrating its potential to solve the problem.

1. Introduction

Tracking pedestrians is the task of localizing each person over time. Some applications require online methods, such as real-time surveillance systems [8] and autonomous vehicles [11]. These algorithms typically utilize the tracking-by-detection (TBD) paradigm, as discussed in a survey by Sun et al. [16]. In this paradigm, an object detector retrieves the pedestrians at each frame. Then, the tracker associates the detections that belong to the same person, assigning them the same identity.

Thus, tracking accuracy is highly dependent on the quality of detections. It is important to note that there may be instances where the detector fails to detect a person's presence. One of the reasons for this is due to occlusions when a person or object blocks the view. In this case, using multiple cameras is helpful to recover persons that are visible from another point of view [10].

Several state-of-the-art techniques [4, 13, 17, 20] train neural networks to track in multiple cameras. However, supervised methods require retraining in new environments, hindering real-world applications.

Lyra et al. [12] proposes an online unsupervised multi-camera tracker. It uses the distance between the given detections to calculate which ones should be paired together. If the pedestrian is not detected in a frame, the tracker will not retrieve his position at that moment. Thus, the tracking algorithm should be capable of handling errors that may still occur.

Detections are typically accompanied by a confidence score, and only those with high scores are utilized. Therefore, using detections with low confidence can decrease the number of missing detections. On the other hand, it elevates the possibility of detecting a person's presence when no one is around. ByteTrack [21] suggests including detections with low confidence and trying to decide which ones are correct during tracking.

StrongSORT [5] proposes some improvements in the classic tracker DeepSORT [19], including a module to recover partially lost tracks. If a person does not appear on time t but is present on times $t-1$ and $t+1$, they interpolate the detections from past and future to create a detection in t and use a Gaussian filter to smooth the coordinates.

As this process uses information from future frames, it occurs offline. Besides, both ByteTrack and StrongSORT were applied only in the single-camera scenario, relying on the view of one camera.

Our work builds upon the proposed module from StrongSORT to recover detections. We adapted this module to become online and to work in a multi-camera environment. Then, we applied it to the multi-camera online tracker proposed by Lyra et al. [12] to evaluate. The tracker and module are unsupervised and, therefore, well-suited for practical applications.

The contributions of this work are:

- An online and multi-camera module to recover missing detections that can be linked to existing trackers (Section 2);
- Quantitative evaluations of the proposed method (Section 3).

2. RetryTRACK

This section explains our module and the complete tracking procedure, which contains four steps and is illustrated in Figure 1. First, we extract detections from the images (Section 2.1). Then, the detections are given to the baseline tracker (Section 2.2). Finally, we insert the module to recover lost detections (Section 2.3) and use a filter to remove duplicates (Section 2.4).

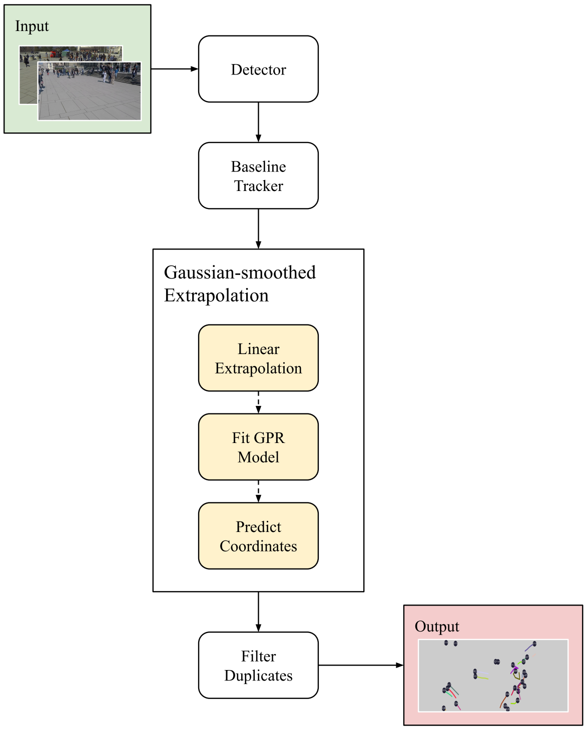


Figure 1. Images from multiple cameras are input to the detector, which retrieves detections as world ground plane coordinates. A baseline tracker computes pedestrian trajectories, and our Gaussian-smoothed extrapolation refines the results, recovering missing detections. The output is the updated pedestrian trajectories.

2.1. Detector

We used the tracker proposed by Lyra et al. [12] as the basis for our method. It relies on a multicamera pedestrian detection proposed by Lima et al. [10]. This detector uses AlphaPose [9] to extract the keypoints located at the pedestrian's ankles, which is used to compute the walker ground point $p_{cam} = (x_{cam}, y_{cam})$. Then, they use the camera calibration to project every pedestrian camera point onto the world ground plane. Each pedestrian has one world ground point per camera. Thus, they gather the closest points and calculate the mean to determine the final pedestrian coordinate in 3D space $p_{world} = (x_{world}, y_{world}, z_{world})$, where all of the pedestrians are on the ground, which is located at $z_{world} = 0$ [10].

2.2. Baseline Tracker

The detections are input to the tracker. Each detection receives an id at the first frame ($t = 0$). In the following frames, they compare the current detections to existing tracks.

A bipartite graph is constructed, with the nodes representing detections from the previous and current frames and edges representing the Euclidean distance between them. If the previous frame detections do not include an older trajectory, a Kalman filter [7] predicts the position of the latest detection in this trajectory. Then, they use the maximum weight matching algorithm to retrieve which detections should be assigned together.

2.3. Gaussian-smoothed Extrapolation

By the end of each frame, we insert the module Gaussian-smoothed Extrapolation (GSE) to recover lost tracks. If a person of the previous frame $t-1$ is not found in the current time t, we use linear extrapolation [2] with the detections of $t-2$ and $t-1$ to generate a new detection at time t using the following equation

$$(x_t, y_t) = (x_{t-2}, y_{t-2}) + f * ((x_{t-1}, y_{t-1}) - (x_{t-2}, y_{t-2})), \tag{1}$$

where

$$f = \frac{f_t - f_{t-2}}{f_{t-1} - f_{t-2}}, \tag{2}$$

f_t is the number of the frame t, and (x, y) is world coordinate.

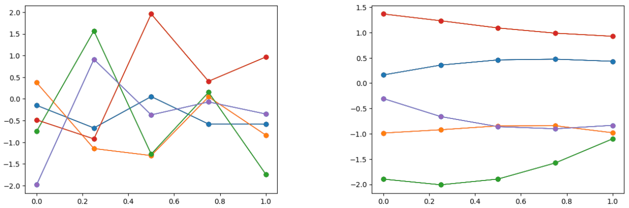
As a stop criterion, we verify if the extrapolated coordinate is still inside the Area of Interest (AOI), a rectangle in the world ground plane in which we want to keep track of pedestrians, defined by $AOI = (x_{min}, x_{max}, y_{min}, y_{max})$. We do not include this coordinate in our results if it is outside the AOI. Thus, we will stop trying to recover this track.

However, the generated points assume the pedestrian's movement to be linear, which is often not the case. Therefore, we use Gaussian Process Regression (GPR) to improve our predictions.

Given a set of observed points, multiple functions can fit them. The GPR considers all possible functions and computes the probability distribution over them. If we draw a straight line between the points, the result will be a noisy function that is inadequate to predict new points. Hence, we use a Radial Basis Function (RBF) kernel to smooth the functions. The kernel can be denoted by

$$k(x_i, x_j) = exp\left(-\frac{\|x_i - x_j\|^2}{2\lambda^2}\right), \tag{3}$$

where x is the frame, and $\lambda = \tau * \log(\tau^3/l)$. l refers to the length of tracks, and τ is set to 10 [5]. Figure 2 exemplifies the effect of the kernel in the functions.



(a) Functions before using RBF.

(b) Functions after using RBF.

Figure 2. Comparison between example functions before and after applying RBF.

Afterwards, the probability distribution [18] is calculated using

$$P(x) = \frac{1}{\sqrt{2\pi\sigma}} exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right), \tag{4}$$

where x is the frame, μ is the mean and σ^2 is the variance.

We fit the GPR model using the frame t as a variable and x_{world} and y_{world} as target values. Thus, the regression function modeled by GPR is

$$P(f|X) = N(f|\mu, K), \tag{5}$$

where X are the frames, f are the x_{world} and y_{world} values, μ is the mean function derived from the probability distribution and K is the kernel function. Then, we predict the smoothed coordinates.

2.4. Filter Duplicates

During experiments, we found that some extrapolated coordinates were too close to another detection. We do not need duplicate occurrences within a short range since two persons can not occupy the same space.

Hence, we discard detections that have less than 0.4m of distance, which is inside the typical width of the shoulders [15].

3. Experiments

This section explains how we evaluated our method and what were the results. Section 3.1 describes the dataset and Section 3.2 the metrics. Then, Section 3.3 compares our technique with the state-of-the-art. Finally, in Section 3.4 we show the ablation study.

3.1. Dataset

We use the public dataset WILDTRACK [3] in our experiments. It has seven cameras with overlapping views and calibration parameters. The cameras have 400 frames with annotations indicating the pedestrian ID, the bounding box at each camera (if the pedestrian is visible from that point of view), and the world ground plane coordinate.

Since other methods need to split the frames for training, we use the last 10% of frames to evaluate our technique. Thus, the test subset is the same for fair comparison.

3.2. Metrics

Three types of errors can occur during tracking. First, a detection can point to an empty space, which is known as a false positive (FP). Second, a real person may not be detected, which is called a false negative (FN). Finally, if a person already being tracked with one ID is assigned to another ID, it is considered a mismatch (MM).

We used the CLEAR MOT metrics [1] multiple object tracking accuracy (MOTA) and multiple object tracking precision (MOTP). While MOTA evaluates the proportion between errors and ground truth objects, MOTP measures the precision of tracking.

Furthermore, we include the common metrics of precision (Prec) and recall (Rcll) [14].

3.3. State-of-the-art

We compare our method with other state-of-the-art techniques in Table 1. Some methods, such as Cheng et al. [4] and Nguyen et al. [13], perform better. However, they require training. Our work is a simple yet effective unsupervised approach and is easier to apply in new environments.

3.4. Ablation

We evaluated each module addition to the baseline tracker. The results are presented in Table 2.

It is possible to see that GSE alone does not improve the results. This is because although it decreases the number of FNs, it increases FPs when extrapolating infinitely. However, including the filter of duplicates, the algorithm performs better than the baseline.

Besides, as the goal was to recover detections, we observed that applying GSE+Filter reduced FNs by 20.3%.

4. Conclusion

In this work, we proposed an online and unsupervised module to recover lost detections. We use linear extrapolation to create new detections and a Gaussian process regressor to smooth the coordinates. Furthermore, we filter close detections to remove duplicates. We evaluate the module's performance using a baseline tracker on the WILDTRACK dataset. This resulted in 79.52% of MOTA, which

	Technique	MOTA	MOTP	FP	FN	MM
Supervised	You & Jiang [20]	74.6%	78.9%	114	107	21
	Vo et al. [17]	75.8%	-	-	-	-
	Cheng et al. [4]	81.6%	81.8%	-	-	-
	Nguyen et al. [13]	97.1%	-	71	7	12
Unsupervised	Lyra et al. [12]	77.1%	94.77%	76	128	**14**
	Ours	**79.52%**	**96.27%**	**76**	**102**	17

Table 1. Comparison with other state-of-the-art techniques in WILDTRACK dataset. We evaluate using the last 10% of frames to compare with supervised methods. Ours is the best unsupervised method.

Technique	MOTA	MOTP	Prec	Rcll	FP	FN	MM
Lyra et al. [12]	77.1%	94.77%	91.56%	86.55%	**76**	128	14
Lyra et al. [12]+GSE	75.95%	95.46%	89.25%	88.97%	102	105	22
Lyra et al. [12]+Filter	76.16%	94.77%	91.16%	85.61%	79	137	**11**
Lyra et al. [12]+GSE+Filter	**79.52%**	**96.27%**	**91.79%**	**89.29%**	**76**	**102**	17

Table 2. Results of each module addition to the baseline tracker in the WILDTRACK dataset. Using GSE with the filter of duplicates is the best result.

is a 2.42% improvement and makes it the best unsupervised method. Furthermore, we were able to reduce the number of FNs by 20.3%.

In future work, we plan to test our technique with other datasets such as MultiviewX [6] to validate its reliability with stronger evidence. Besides, we will analyze its limitations by using different numbers of cameras and automatic calibration, as the need for calibrated cameras can also be a drawback for real applications.

References

[1] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 3

[2] Claude Brezinski and M Redivo Zaglia. *Extrapolation methods: theory and practice*. Elsevier, 2013. 2

[3] Tatjana Chavdarova, Pierre Baqué, Stéphane Bouquet, Andrii Maksai, Cijo Jose, Timur Bagautdinov, Louis Lettry, Pascal Fua, Luc Van Gool, and François Fleuret. Wildtrack: A multi-camera hd dataset for dense unscripted pedestrian detection. In *CVPR*, pages 5030–5039, 2018. 3

[4] Cheng-Che Cheng, Min-Xuan Qiu, Chen-Kuo Chiang, and Shang-Hong Lai. Rest: A reconfigurable spatial-temporal graph model for multi-camera multi-object tracking. In *ICCV*, pages 10051–10060, 2023. 1, 3, 4

[5] Yunhao Du, Zhicheng Zhao, Yang Song, Yanyun Zhao, Fei Su, Tao Gong, and Hongying Meng. Strongsort: Make deepsort great again. *IEEE TMM*, 25:8725–8737, 2023. 1, 3

[6] Yunzhong Hou, Liang Zheng, and Stephen Gould. Multiview detection with feature perspective transformation. In *ECCV*, 2020. 4

[7] Rudolph Emil Kalman. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82 (1):35–45, 1960. 2

[8] Mingwei Lei, Yongchao Song, Jindong Zhao, Xuan Wang, Jun Lyu, Jindong Xu, and Weiqing Yan. End-to-end network for pedestrian detection, tracking and re-identification in real-time surveillance system. *Sensors*, 22(22), 2022. 1

[9] Jiefeng Li, Can Wang, Hao Zhu, Yihuan Mao, Hao-Shu Fang, and Cewu Lu. Crowdpose: Efficient crowded scenes pose estimation and a new benchmark. In *CVPR*, pages 10863–10872, 2019. 2

[10] Joao Paulo Lima, Rafael Roberto, Lucas Figueiredo, Francisco Simoes, and Veronica Teichrieb. Generalizable multi-camera 3d pedestrian detection. In *CVPR*, pages 1232–1240, 2021. 1, 2

[11] Yongqiang Lu, Hongjie Ma, Edward Smart, and Hui Yu. Real-time performance-focused localization techniques for autonomous vehicle: A review. *IEEE Transactions on Intelligent Transportation Systems*, 23(7):6082–6100, 2022. 1

[12] Victor Lyra., Isabella de Andrade., João Lima., Rafael Roberto., Lucas Figueiredo., João Teixeira., Diego Thomas., Hideaki Uchiyama., and Veronica Teichrieb. Generalizable online 3d pedestrian tracking with multiple cameras. In *Proceedings of the 17th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications - Volume 5: VISAPP,*, pages 820–827. INSTICC, SciTePress, 2022. 1, 2, 4

[13] Duy M. H. Nguyen, Roberto Henschel, Bodo Rosenhahn, Daniel Sonntag, and Paul Swoboda. Lmgp: Lifted multicut meets geometry projections for multi-camera multi-object tracking. In *CVPR*, pages 8866–8875, 2022. 1, 3, 4

[14] David L Olson and Dursun Delen. *Advanced data mining techniques*. Springer Science & Business Media, 2008. 3

[15] Nirajan Shiwakoti, Majid Sarvi, and Martin Burd. Using non-human biological entities to understand pedestrian

crowd behaviour under emergency conditions. *Safety Science*, 66:1–8, 2014. 3

[16] Zhihong Sun, Jun Chen, Liang Chao, Weijian Ruan, and Mithun Mukherjee. A survey of multiple pedestrian tracking based on tracking-by-detection framework. *IEEE TCSVT*, 31 (5):1819–1833, 2020. 1

[17] Minh Vo, Ersin Yumer, Kalyan Sunkavalli, Sunil Hadap, Yaser Sheikh, and Srinivasa G. Narasimhan. Self-supervised multi-view person association and its applications. *IEEE TPAMI*, 43(8):2794–2808, 2021. 1, 4

[18] Jie Wang. An intuitive tutorial to gaussian process regression. *Computing in Science & Engineering*, 25(4):4–11, 2023. 3

[19] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, pages 3645–3649, 2017. 1

[20] Quanzeng You and Hao Jiang. Real-time 3d deep multi-camera tracking. *arXiv preprint arXiv:2003.11753*, 2020. 1, 4

[21] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *ECCV*, pages 1–21, Cham, 2022. Springer Nature Switzerland. 1