Person Tracking in Heavy Industry Environments with Camera Images

Nico Zengeler[1][0000−0002−1319−5877], Alexander Arntz[1], Dustin Keßler[1], Matthias Grimm[1], Ziyaad Qasem[1], Marc Jansen[1], Sabrina Eimler[1], and Uwe Handmann[1]

Hochschule Ruhr West
Lützowstraße 5
46236 Bottrop, Germany

{nico.zengeler, alexander.arntz, dustin.kessler, matthias.grimm, ziyaad.qasem, marc.jansen, sabrina.eimler, uwe.handmann}
@hs-ruhrwest.de

https://www.hochschule-ruhr-west.de/

Abstract. In this paper, we propose a method to localise and track persons in heavy industry environments with multiple cameras. Using the OpenPose network, we localise the persons feet points on each cameras image individually and perform according 3D transformations. With prior knowledge about the camera settings in the environment, we use a rule-based system to assess which sensor detections to fuse. We then apply Kalman filtering in order to stabilise the tracking. Due to a variable image stack size, our method may increase accuracy if provided with additional computational resources by processing more frames in real-time. We have simulated a heavy industry scenario and use the recorded video material and position data as a basis for our evaluation.

Keywords: heavy industry · Industry 4.0 · person tracking · artificial intelligence · image processing

1 Introduction

Industry 4.0 requires software to support and enhance existing heavy industry structures. Digital facility management systems help to increase productivity, work safety and the production process transparency [12, 2, 14, 10]. For example, concerning work safety, in a case of emergency, workers may want to follow the shortest route to the exit. In a steel industry site, this route may lead across a potentially dangerous area, for example a freshly rolled, hot sheet steel. Moving on that sheet would cause the worker's boots to melt with the hot steel, which causes a high impact on the workers health and high costs for the corporation. An intelligent work safety system may prevent such a situation by providing the right warning at the right time using appropriate means.

Such a system may use cameras and mobile devices to locate and identify persons in dangerous situations. To do so, such systems need to gather and analyse as much information about the production process as possible in realtime, put them into context and perform the right actions. The project *DamokleS 4.0* [5] aims to develop a system to support employees in heavy industry using modern hardware and software. In this particular contribution, we use camera images to perform human foot point localisation and provide these detections to a rich context model. Knowing the worker's locations and roles within the production process allows the context model to display valuable, individual information. In our scenarios, augmented reality (AR) devices poll the context model in order to show supportive advises to their current user. For example, an information service provides relevant data about the workers current task to his or her augmented reality device, as shown in figure 1. This way a worker receives context-sensitive and role-specific information, for example the state of a machine, a concrete work instruction or an evacuation route in case of emergency.

Fig. 1: Exemplary augmented reality information depending on the workers current position. Left: an instruction to check a serial number. Right: displaying an evacuation route in case of emergency.

In this paper we describe a holistic software approach to localise and track workers in heavy industry settings, solely on the basis camera images, using methods of artificial intelligence. We begin this paper by delineating this contribution within the context of the project *DamokleS 4.0* and presenting the current state of the art. In section 3, a description of our software implementation explains the workflow of our program in detail. Then we present the setup of our laboratory experiments and examine how we have collected our data in section 4. We also statistically evaluate the accuracy of our system compared to ground truth data, as provided by the augmented reality devices positioning. In the last section we conclude with a short discussion of our final results, the pros and cons of our implementation and possible future work.

2 State of the art

Current state of the art research investigates person tracking techniques under various points of view with different approaches. For example, [20] proposes a person tracking algorithm for an autonomous unmanned aerial vehicle. In this approach, a drone with a surveillance camera follows individual persons, which allows for a very flexible surveillance system with the benefit of easy face recognition. Compared to a stationary camera-based approach, the aerial vehicle seems impractical for an industrial application, as the flying drone may collide with moving objects like cranes, vehicles or even other persons. Considering person identification techniques, state of the art researchers focus on methods of deep transfer learning [7, 4]. These methods allow for single-shot person re-identification and prove that transfer learning may increase detection performance in that domain using very little training data. For our heavy industry application, we chose to identify workers via their wearable smart devices instead of face recognition. To handle the problem of tracking multiple persons, [9] utilise slow feature analysis [23]. [1] presents a computational framework for interpreting person tracking data, which consists of four modules for tracking instantaneous and short-time features as well as unsupervised and supervised machine learning techniques for higher levels of abstraction.
Concerning the *DamokleS 4.0* project [5], [17] describes the overall software architecture underlying our context model [6]. Also, [17] sketched the essential ideas that drive our test scenarios as well as the associated processes for implementation in mobile devices. The suggested scenarios concern workplace safety as well as production and maintenance applications. The proposed approach provides context-based support for factory employees during all these scenarios. For context recognition, [17] proposes the usage of mobile device sensors and external sensors devices mounted in the factory building, for example cameras and beacons. [24] evaluated a variety of human detection methods and concluded that the OpenPose system [3, 18, 21] suits our purpose best as it provides a most reliable foot point detection, even under challenging image conditions.
In a related project we developed a video surveillance system to protect critical infrastructures [8]. In this project we designed a software architecture that supports human operators to detect, track and recognize suspicious subjects in case of an alert. The human operator may sort video frames by personally selecting important features. He or she may flag suspicious subjects and reidentify them in a video database. The camera-based data analysis consisted of several image processing modules like a salient-based people detection and a histogram of oriented gradients (HOG) algorithm based on the implementation of [16]. We decided to use a GPU-based implementation to speed up the HOG algorithm and fulfill our realtime requirements. The scenarios described in [8] resemble those in the context of heavy industries with respect to challenges introduced by different light conditions and the high need for fast algorithms.
On the basis of the referenced developments, we can state that the interaction of the collected data and the constantly evolving algorithms holds a great potential for the improvement of industrial processes and the everyday working life.

3 Implementation

Our software architecture consists of three different modules, which operate on a live video stream of multiple cameras in real-time. Figure 2 shows the program flowchart of the single processing steps. Starting with a human foot point detection system, for which we have used the OpenPose architecture [3, 18, 21], a coordinate transformation from image coordinates into world coordinates provides input to the second module, a rule-based sensor fusion approach. The rule-based system also prepares the trajectories for the third module, a Kalman filter, by assigning them to linear tracks. All processing steps take place on the same stack of images, guaranteeing real-time capability in a trade-off between stack size and available computing resources. The more images the software sees, the more accurate it gets. Maintaining real-time capability only depends on the available computing resources. As input we present a stack of k images per camera and choose k such that the program runs as fast and as accurate as possible. Increasing k leads to more accurate detections at the cost of higher computation time. As output we obtain current person locations in world coordinates, which we may send to a remote context model. The context model may relate the locations with other data, for example to identify persons via smart devices.

3.1 Foot point detection and coordinate transformation

Before performing the foot point detection, we improve the camera images by performing an adaptive histogram equalisation with a tile grid size of eight by eight pixels [15]. We then localise the person's foot points in camera image coordinates using the corresponsing foot keypoints of the COCO model as provided by OpenPose [3, 18, 21]. Our calibration process assumes that the person moves on a flat plane, so we use a temporary constant height coordinate $z = 1$, which we simply discard after transformation. To transform the camera coordinates $(x, y, 1)^T$ into world coordinates $(p_x, p_y)^T$, we use the intrinsic camera matrix M, the rotation matrix R and the translation vector d, which we have obtained via a standard calibration process using chessboard patterns [13, 11]. Constructing an auxiliary matrix

$$R' = \begin{pmatrix} R_{0,0}, R_{0,1}, R_{0,2} \\ R_{1,0}, R_{1,1}, R_{1,2} \\ d_0, d_1, d_2 \end{pmatrix} \tag{1}$$

leads to a coordinate transformation that reads as follows:

$$\begin{pmatrix} p_x \\ p_y \end{pmatrix} = \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} (R'M)^{-1} \tag{2}$$

The resulting world coordinates relate to the origin of the chessboard pattern. For multiple cameras, which observe distinct parts of the environment, we perform multiple extrinsic calibrations and then translate the world coordinates by the distance vectors between the different coordinate origins.

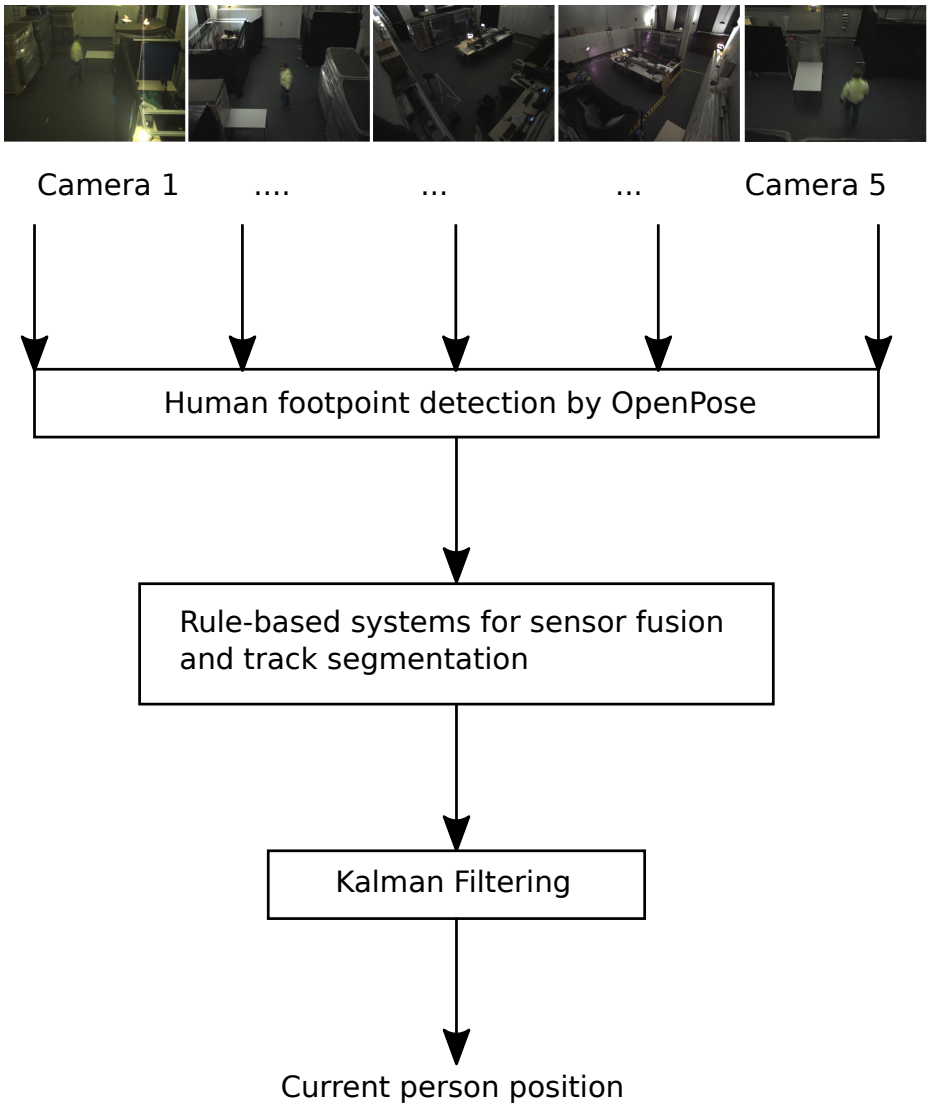


Fig. 2: The program flowchart of our person tracking procedure. The top row shows example frames, taken from the same trial at the same time. From left to right: cameras $C1$ to $C5$ as shown in figure 3. The person currently moves within sight of cameras $C1, C2, C5$ but out of sight for cameras $C3$ and $C4$.

3.2 Sensor fusion and track separation

The second module consists of rule-based systems, that start by fusing the detections from the localisation module using prior knowledge about the camera setup. To solve the problem of missing detections by noise, we perform an autocompletion within the k frames: if in a frame we find no detection, but in the previous and following frame we do, we replace the missing detection with the geometric mean between the two successful detection. This way, we complete the detections within k frames and maintain real-time applicability for an optimal value of k. Knowing on which frame we have a detection, we put these detections into a two-dimensional boolean matrix, which tells us about which camera yields a coherent detection within the real-time window. Using this matrix, we apply a rule-based system that decides which detections to fuse together. To do so, a hard coded rule set reflects our prior knowledge about the concrete camera setup in the environment. Upon this knowledge, we apply a set of conditional clauses to decide the world coordinate fusion. If, for example, $C3$ and $C4$ detect the same person, we calculate a geometric mean of the two proposed world coordinates. To prepare the position data for Kalman filtering, we apply a rule set that assigns each position to a unique track. Each track features a steady motion, which simplifies the Kalman filtering procedure.

3.3 Kalman filtering

In order to smooth the resulting trajectory, we employ a Kalman filter [22] on each of the separated tracks, as shown in figure 5. To initialize the Kalman filter, we use a four-dimensional steady motion dynamics, capturing the persons position $(p_x, p_y)^T$ as well as the persons current velocity $(v_x, v_y)^T$ with respect to a fixed time step dt as given by the camera recording frequency:

$$F = \begin{pmatrix} 1 & 0 & dt & 0 \\ 0 & 1 & 0 & dt \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

such that:

$$\begin{pmatrix} p_x \\ p_y \\ v_x \\ v_y \end{pmatrix}_{t+1} = F \cdot \begin{pmatrix} p_x \\ p_y \\ v_x \\ v_y \end{pmatrix}_{t} \tag{4}$$

We use a unit matrix to initialise the Kalman filter covariance matrix. For the estimation process, we iterate over k subsequent positions, thus maintaining real-time applicability.

4 Evaluation

We use the recordings of a laboratory study to evaluate our person tracking approach in a simulated industrial environment. In this study, as part of the *DamokleS 4.0* project [5], the test persons wear augmented reality glasses which guide them through a parcour. During this course, they have to solve three tasks and, for the last part, follow an evacuation route to the exit, as depicted in figure 3. The original user study featured two different navigation modalities and corresponding questionnaires, which aimed to evaluate the test person's feelings and attitudes towards this technology from a psychological point of view. For our person tracking study, we discard this information and merely use the collected video recordings.

4.1 Setup

Figure 3 shows the setup of our test course and the camera positions. As shown in the example frames in figure 2, the test persons wear safety vests and move in sight of a certain subset of our cameras. As shown in figure 3, the camera sets $(C1, C2, C5)$ and $(C3, C4)$ observe distinct parts of the environment. Each set has a unique calibration origin, which we relate to the parcour starting point by a translation vector which we have measures using a scale.

4.2 Results

Our cameras recorded video footage with eight to twelve frames per second, so we have used a stack size of $k = 4$ frames to maintain real-time applicability with our hardware. As the augmented reality device recorded positions with a rate of two positions per second, the cameras yield more data in the same time as they record with a higher frame rate. Figure 5 shows the complete trajectories for the ground truth data as provided by the augmented reality device, the raw camera position estimations after sensor fusion, the separated tracks and the final positions after Kalman filtering. The statistics about the travelled distances, durations and velocities, as shown in figure 6, ignore the different temporal resolutions induced by different recording rates. In order to compare the mean deviations between the estimated camera positions and the ground truth trajectories, as shown in figure 4, we solve the problem of the different temporal resolutions by searching the nearest point in the ground truth positions for each camera position.
The average of the mean deviations between the raw camera position estimates and the ground truth data evaluates to about $0.67m$, while the average of the mean deviations between the Kalman filtered final positions and the ground truth positions lies slightly lower at about $0.59m$. This corresponds to the trajectories, which come closer to the ground truth after Kalman filtering. Looking at the trajectories in figure 5, we find a slight metrical distortion in the start and end region, as only camera $C2$ observes this region. Analysing the histograms in figure 6, we can state that both the travelled distances and the average velocities

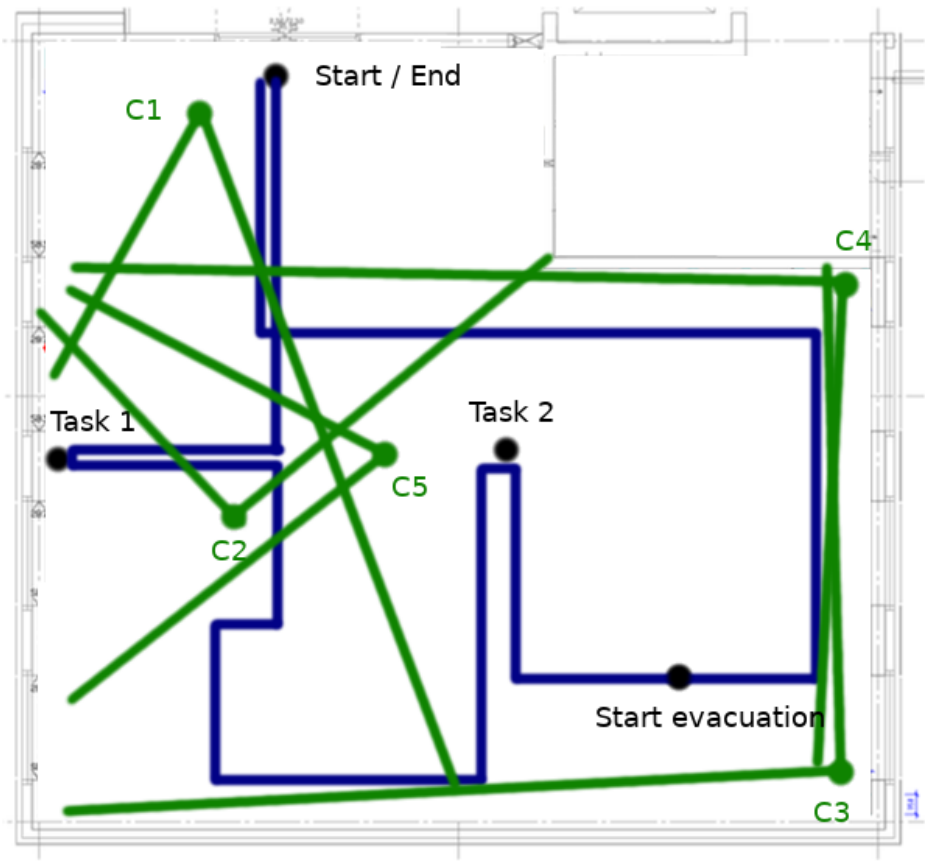


Fig. 3: The test parcour of our simulated industrial environment. The test persons follow a route (blue) and solve a number of tasks. Five cameras (green) record video footage.

come closer to the original distribution after Kalman filtering. Furthermore, we can see that in the evacuation tracks, tracks number nine to twelve, the average velocities show higher values.

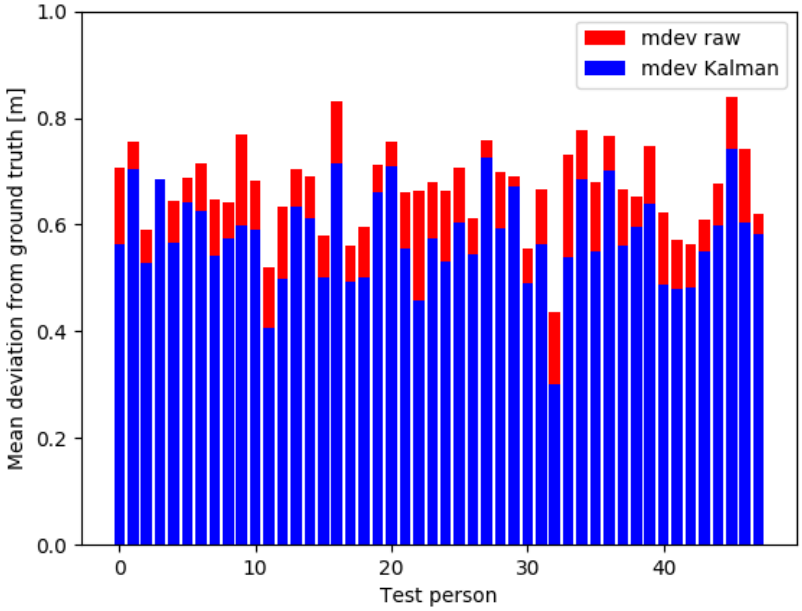
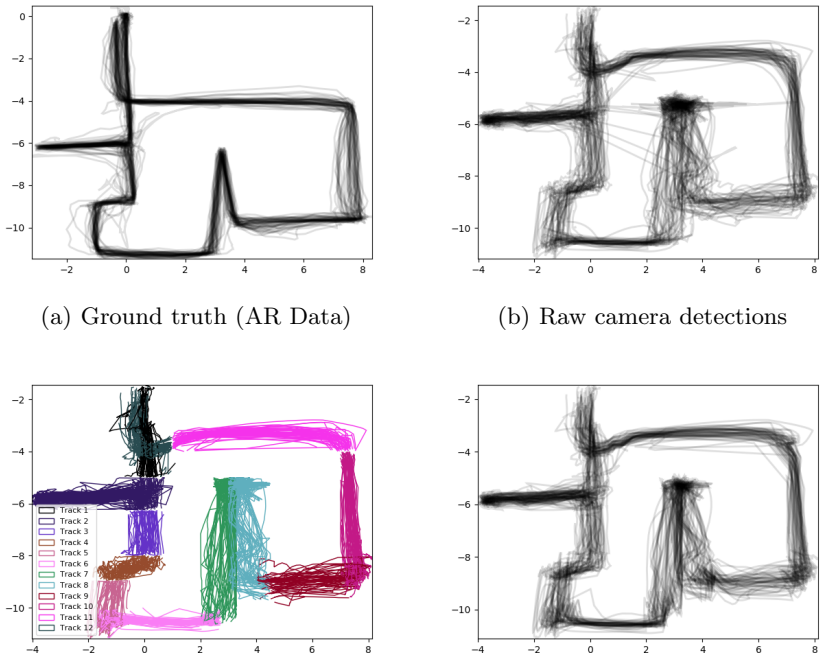


Fig. 4: The mean deviation in meters between the estimated camera positions and the ground truth positions from the augmented reality device for each test person. The red bars show the mean deviations for the raw camera estimations, and the blue bars show the mean deviations for the Kalman filtered positions.

5 Conclusion

We have contributed a method to evaluate person detection models for heavy industry environments and published our source code, raw data and results under [19]. From the metrical distortion that we have observed and described in section 4.2, we conclude that for a reliable location estimation a person must move in sight of at least two calibrated cameras. To obtain accurate tracking results on camera images, we strongly recommend the usage of additional means like Kalman filters. For reasons of data protection, we decide to identify persons via their smartphones in a context model instead of using face recognition on camera images.



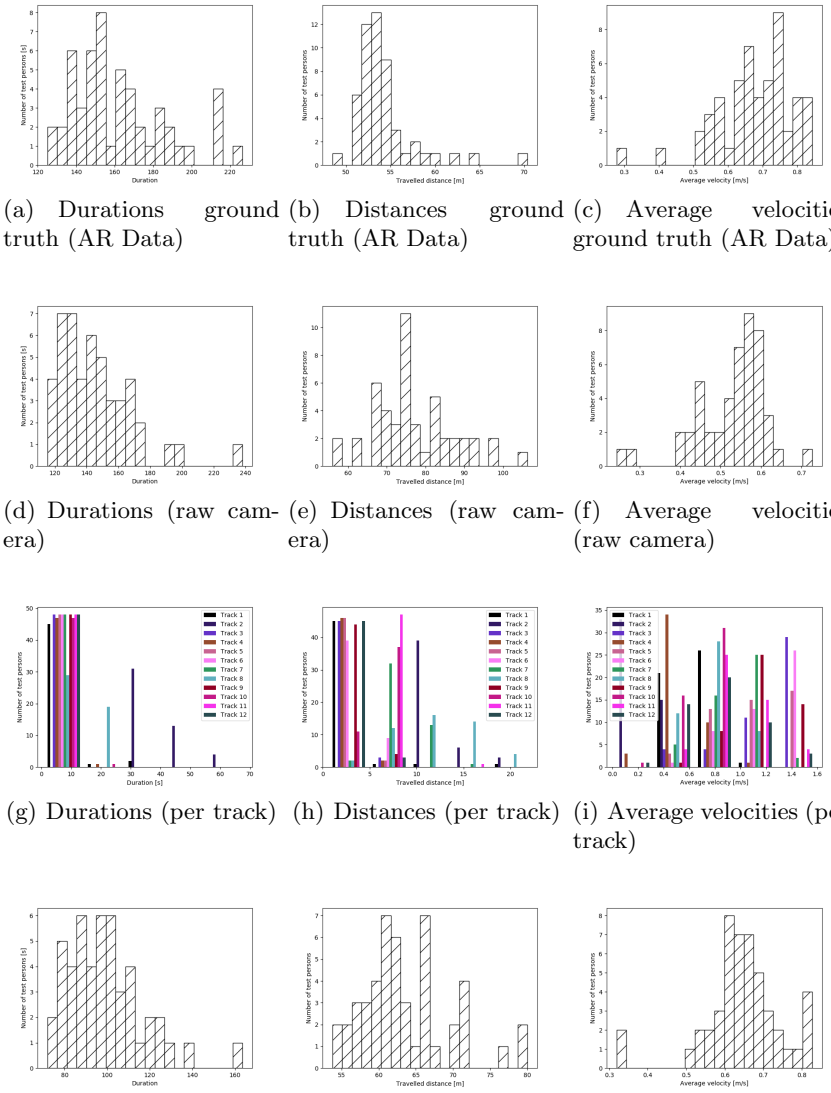
(a) Ground truth (AR Data)

(b) Raw camera detections

(c) Track segmentation

(d) Kalman filtered

Fig. 5: Resulting trajectories, all scales in meters. The top left figure shows the ground truth as provided by the augmented reality device, the top right plot demonstrates the trajectories after the detection of foot points and the appliance of the first rule-based system. In the bottom left plot we visualise the results after track separation. As shown in the bottom right plot, the trajectories after Kalman filtering closely resemble the positions from the ground truth data.



(a) Durations ground truth (AR Data)
(b) Distances ground truth (AR Data)
(c) Average velocities ground truth (AR Data)
(d) Durations (raw camera)
(e) Distances (raw camera)
(f) Average velocities (raw camera)
(g) Durations (per track)
(h) Distances (per track)
(i) Average velocities (per track)
(j) Durations (Kalman filtered)
(k) Distances (Kalman filtered)
(l) Average velocities (Kalman filtered)

Fig. 6: The histograms showing overall statistics about the results after our image processing procedure. From left to right: duration in seconds, travelled distances in meters, average velocities in meters per second. From top to bottom: ground truth data from the augmented reality (AR) device, raw camera detections, statistics per individual track and the final Kalman filtered results.

5.1 Discussion

In our evaluations we used the default parameters for of all our third party software, like the OpenPose framework and the Matlab camera calibration toolbox. Changing these hyper parameters may improve results. More computational resources allows our approach to deliver better tracking results while maintaining real time capability by increasing a single parameter, the image stack size. The sensor fusion in form of a rule-based system relies on previous knowledge but allows for easy changes due to its transparent rule set. Our model assumes that the persons move on a flat plane, so it can't tell different height levels from each other. The rule-based systems, although transparent to the user and easy to change, miss the flexibility to simply work for other setups. The same problem arises for our rather static track segmentation. Our laboratory study only provided video material containing one single person in the parcour, so we did not evaluate our system for multiple persons. Our systems makes no assumptions on the number of persons, which we leave for future investigations.

5.2 Future Work

To further develop our approach, a more flexible track assignment might yield a high profit. For example, a reinforcement learning agent may learn to open and close track assignments dynamically. Also we might employ Kalman filters on the detections in image coordinates to further stabilise the detections. Using means of Transfer Learning, we may investigate how to easily adapt our models to other situations.

Acknowledgement

This work was supported by the *DamokleS 4.0* project [5] project funded by the European Regional Development Fund (ERDF), the European Union (EU) and the federal state North Rhine Westphalia.

References

1. Amin, S. & Burke, J. *OpenMoves: A System for Interpreting Person-Tracking Data* in (). doi:10.1145/3212721.3212846.
2. Bunte, A. *et al.* Evaluation of Cognitive Architectures for Cyber-Physical Production Systems. *CoRR* **abs/1902.08448.** arXiv: 1902.08448. http://arxiv.org/abs/1902.08448 (2019).
3. Cao, Z., Simon, T., Wei, S.-E. & Sheikh, Y. *Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields* in *CVPR* (2017).
4. Chen, H. *et al. Deep Transfer Learning for Person Re-Identification* in (). doi:10.1109/BigMM.2018.8499067.
5. *DamokleS 4.0 - IKT für Cyber Physical Systems* https://www.damokles40.eu/. Accessed: 2018-10-31.

6. Dey, A. K. Understanding and Using Context. *Personal Ubiquitous Comput.* **5,** 4–7. ISSN: 1617-4909 (Jan. 2001).
7. Gómez-Silva, M., Izquierdo, E., de la Escalera, A. & Armingol, J. M. Transferring learning from multi-person tracking to person re-identification. *Integrated Computer-Aided Engineering,* 1–16 (Apr. 2019).
8. Handmann, U., Hommel, S., Grimm, M. & Malysiak, D. APFel - Fast multi camera people tracking at airports, based on decentralized video indexing. **Science**^{3} $--Safetyand Security$**,** 48–55 (Jan. 2014).
9. Hao, T., Wang, Q., Wu, D. & Sun, J. Multiple person tracking based on slow feature analysis. *Multimedia Tools and Applications* **77.** doi:10.1007/s11042-017-5218-4 (Sept. 2017).
10. Hasselbring, W. *et al.* Industrial DevOps. *CoRR* **abs/1907.01875.** arXiv: 1907.01875. http://arxiv.org/abs/1907.01875 (2019).
11. Heikkila, J. & Silven, O. *A Four-step Camera Calibration Procedure with Implicit Image Correction* in *Proceedings of the 1997 Conference on Computer Vision and Pattern Recognition (CVPR '97)* (IEEE Computer Society, Washington, DC, USA, 1997), 1106–. ISBN: 0-8186-7822-4. http://dl.acm.org/citation.cfm?id=794189.794489.
12. Hermsen, K. *et al. Dynamic, Adaptive and Mobile System for Context-Based and Intelligent Support of Employees in the Steel Industry* in *4th ESTAD (European Steel Technology and Application Days)* (Düsseldorf, Germany, 2019). https://www.metec-estad2019.com/files/190619_metec-estad_programmflyer_a5q_web-5.pdf.
13. MathWorks. *Camera Calibration* https://mathworks.com/. Feb. 23, 2018.
14. Nouiri, M., Trentesaux, D. & Bekrar, A. EasySched: a multi-agent architecture for the predictive and reactive scheduling of Industry 4.0 production systems based on the available renewable energy. *CoRR* **abs/1905.12083.** arXiv: 1905.12083. http://arxiv.org/abs/1905.12083 (2019).
15. Pizer, S. M. *et al.* Adaptive Histogram Equalization and Its Variations. *Comput. Vision Graph. Image Process.* **39,** 355–368. ISSN: 0734-189X (Sept. 1987).
16. Prisacariu, V. & Reid, I. *fastHOG - a real-time GPU implementation of HOG* tech. rep. 2310/09 (Department of Engineering Science, 2009).
17. Qasem, Z., Bons, J., Borgmann, C., Eimler, S. & Jansen, M. *Dynamic, Adaptive, and Mobile System for Context-Based and Intelligent Support of Employees in Heavy Industry* in (). doi:10.1109/ES.2018.00021.
18. Simon, T., Joo, H., Matthews, I. & Sheikh, Y. *Hand Keypoint Detection in Single Images using Multiview Bootstrapping* in *CVPR* (2017).
19. *Source code and data used in this paper* https://gitlab.hs-ruhrwest.de/nico.zengeler/detectionprocessing. Accessed: 2019-06-15.
20. Surinta, O. & Khruahong, S. *Tracking People and Objects with an Autonomous Unmanned Aerial Vehicle using Face and Color Detection* in (). doi:10.1109/ECTI-NCON.2019.8692269.
21. Wei, S.-E., Ramakrishna, V., Kanade, T. & Sheikh, Y. *Convolutional pose machines* in *CVPR* (2016).

22. Welch, G., Bishop, G., *et al. An introduction to the Kalman filter*
23. Wiskott, L. & Sejnowski, T. J. Slow Feature Analysis: Unsupervised Learning of Invariances. *Neural Computation* **14,** 715–770 (2002).
24. Zengeler, N. *et al. An Evaluation of Human Detection Methods on Camera Images in Heavy Industry Environments* in *2019 IEEE 14th Conference on Industrial Electronics and Applications (ICIEA)* ().