A Multi Modal People Tracker for Real Time Human Robot Interaction

Tim Wengefeld, Steffen Müller, Benjamin Lewandowski and Horst-Michael Gross

***Abstract*— Tracking people in the surroundings of interactive service robots is a topic of high interest. Even if image based detectors using deep learning techniques have improved the detection rate and accuracy a lot, for robotic applications it is necessary to integrate those detections over time and over the limited ranges of individual sensors into a global model. That data fusion enables a continuous state estimation of people and helps reducing the false decisions taken by individual detectors and increasing the overall range. In this paper, we present a tracking framework with a new distance measure for data association and a proper consideration of individual sensors' accuracies. By means of that, we could deal with high false detection rates of laser-based leg detectors without introducing further heuristics like a background model. The proposed system is compared to other tracking approaches from the state of the art. Furthermore, we present a novel manually annotated benchmark dataset for multi sensor person tracking from a moving robot platform in a guide scenario, which will be made publicly available.**

I. INTRODUCTION

Complex scenarios [1, 2] in which robots interact with humans require complex approaches in order to perceive people in the robot's environment. While the pure task for detection seems to be solved due to recent deep learning approaches [3, 4], the task of how to combine these information in world coordinates to consistent tracks over time, in order to enable complex HRI tasks, remains still an open issue. Tracking is particularly interesting when it comes to the fusion of different types of sensors, which cover different purposes and perception ranges.

Our target scenario is a robot which operates in a retail store[1] during opening hours to scan shelves for sold out goods and guides interested customers to special offers. Therefore, it has to be aware of people in its vicinity in order to navigate politely. The robotic platform is equipped with several sensors for different perception tasks (see Fig. 2). The three fish-eye cameras for example cover a large field of view to detect persons even when standing close to the robot e.g. for interaction tasks. However, they are unsuitable to detect persons far away due to their low resolution. The Kinect2 on the other hand needs a minimum distance to capture whole persons but its high resolution data are suitable to detect people at large distances with the disadvantage of a smaller opening angle.

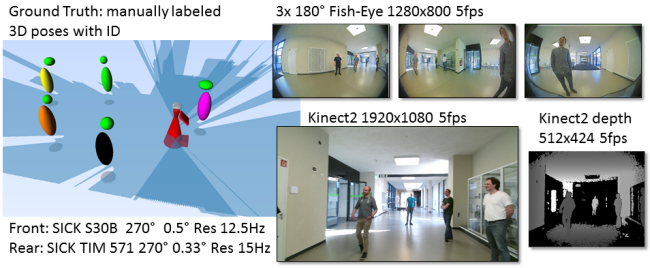

Fig. 1: A snapshot from our dataset with all the data channels recorded. Colored blobs are the ground truth labels, while the green blobs are the results of our tracker focusing on the people's heads

Two 2D lidar sensors for navigation can also be used for detecting peoples' legs. They cover a large area of up to ten meters in horizontal plane at foot height over ground. Unfortunately, detectors on laser data in cluttered environments tend to yield a huge amount of false positive detections, but the fast update rates allow a continuous update of the peoples' positions. This is crucial for a socially acceptable navigation in order to react on fast changes of peoples' moving direction.

Laser-based leg detectors but also image-based detectors mostly provide a score or significance describing how certain a detection is. Often these information are only used by means of thresholding uncertain detections. In contrast, we suggest to use the scores and statistics gained from a test dataset, in order to model the probability of a tracker hypothesis to represent a real person or not. This also concerns the way new hypotheses are spawned in the tracker and helps combining different sensors with different characteristics.

In this paper, we present a generic framework which is able to integrate all these information in an unified manner. Therefore, we are able to provide a compact representation of a robots surroundings for subsequent HRI and navigation modules which take sensor/detector specific update rates and uncertainties in the detections into account. Even though, this work describes a system bound to our robotic platform, we expect to give valuable input for other multi sensor systems which face the same issues described above.

The contributions of this paper are:

- a modular tracking framework for integrating various detectors working on different types of sensor data,
- a model for the probability of a hypothesis to represent a person at all,

All Authors are with Neuroinformatics and Cognitive Robotics Lab, Technische Universitaet Ilmenau, 98694 Ilmenau, Germany. tim.wengefeld@tu-ilmenau.de

This work has received funding from the German Federal Ministry of Education and Research (BMBF) to the project 3D-PersA2 (grant agreement no. 03ZZ0460), and to the project ROTATOR (grant agreement no. 03ZZ0437D) both in the program Zwanzig20 Partnership for Innovation as part of the research alliance 3DSensation.

[1]https://www.tu-ilmenau.de/neurob/projects/rotator/

- a comparison of the trackers performance to other state of the art systems,
- a new benchmark dataset[2], which consists of data from 4 cameras and 2 lidar scanners from a moving platform during a guide scenario which will be made publicly available as ROS bags,

II. RELATED WORKS

Tracking multiple persons, within single or multi sensor setups, has undergone extensive research over the last several decades, mainly in the vision and robotic community. We follow the categorization of the survey from [5] which divides tracking into online approaches, i.e. only using sensor data from the past for the estimation and offline approaches which process a batch of sensor readings. While the purpose is the same, the targeted applications preserve different constraints. Vision scenarios, like surveillance, typically make use of offline approaches, because they are less real time constrained and allow delayed results to better handle ambiguities. Robotic HRI applications on the other hand are bound to immediate results since the dynamic fast changing environments have impact on the robots navigation and HRI behavior and therefore often rely on online approaches. As shown in [6], different approaches cannot be deployed out of the box for each scenario. There, more elaborated offline tracking approaches like [7] from the vision community perform on par with simple online filters [8] when parameterized to deliver real time results. Therefore, we constrain the rest of this section to tracking approaches which have already been deployed in dynamic robotic scenarios with a multiple sensor setup like ours.

A. Multi Sensor Tracking

Approaches for tracking persons from multiple sensor inputs mainly originate from the robotic community. In [9] an approach was presented which uses estimations from a leg and a face detector and compared different filter approaches (EKF, UKF, particle filter). [8] used laser leg detections and an upper body depth template detector deploying the same tracking back-end as [9]. Volkhardt et al. [10] tested different combinations of visual face, upper- and full-body detectors in combination with a leg detector as input for a Kalman Filter. All of these approaches have in common that they fuse laser detections for a wider range of view with vision based detectors with better detection qualities. However, all of them just cover a more or less restricted area at the front of the robot and not a real 360° tracking like we are aiming for.

B. Track Initialization Logic

Two of the core problems for all tracking approaches are the correct handling of false positive detections and the introduction of new hypotheses in the tracker. Some approaches [8, 9] use leaky counters for the number of detections that support a hypothesis or just insert new tracks if a certain motion profile is observed [11]. Other approaches [6, 10] insert new tracks immediately, but only consider them as certain if they are confirmed by more than one detector. One conclusion from [6], where different tracking approaches have been compared, was that in a highly dynamic environment there is no optimal solution to handle these problems for every field of application. A liberal strategy to insert new hypotheses could lead to freeze the robots social navigation behavior while a too conservative one will undermine the social acceptance due to misses. Furthermore, deleting hypotheses too fast could lead to an interrupted HRI behavior.

[2]https://www.tu-ilmenau.de/neurob/data-sets-code/nikr-tracking-data-sets/

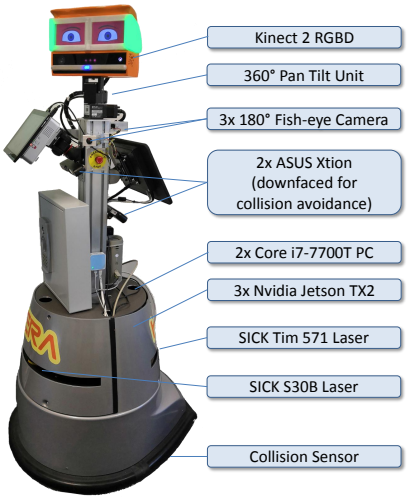

Fig. 2: Used robot platform with its sensors and processing hardware.

C. Conclusion

Even though filter based tracking approaches for person tracking exist for over two decades of research, they are still standard when it comes to a real-world application. Therefore, we focus on how such a filter approach can be used in a complex sensor setup with six sensors covering different detection areas around the robot and various data representations (laser, RGB and depth images). Moreover, we will show how such an approach can be extended with a proper probabilistic track initialization logic and false positive handling.

III. TRACKING SYSTEM

A. Overview of the Tracking System

As already described, our system has various sensors each producing RGB images, depth images, or range scans at distinct time stamps asynchronously (see blue box in Fig. 3). These data are processed by various detection modules (green box) each providing person detections in the sensor space with individual detection scores that can be used for modeling the probability of a hypothesis to actually represent a person. In following feature extraction modules, these detections are converted to unified 3D position hypotheses in

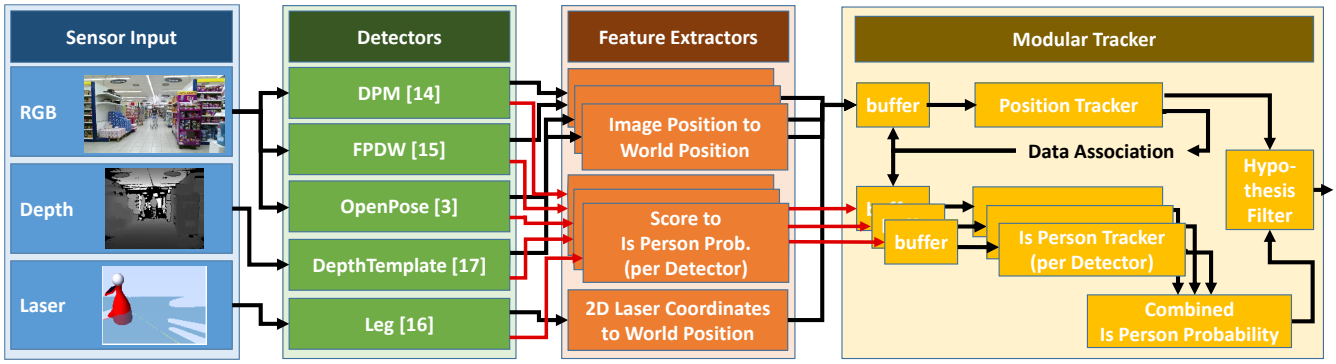

Fig. 3: Overview of our tracking system with individual submodules divided into four categories. Input sensor data (blue) is processed by sensor specific detection modules (green) which deliver detections in sensor coordinates. Feature extractors (orange) take these detections and compute the position in world coordinates and convert the detection scores (red lines) into probabilities for detections to represent an actual human. In the Tracker (yellow) all observations are buffered and sorted by time to handle varying processing times of previous modules. The position tracker's hypotheses are used for data association (which observation belongs to which hypothesis) and existence probabilities are estimated for each detection cue independently. Finally, the combined probability is used to filter out weak hypotheses.

form of 3D Gaussian distributions that can be tracked over time later on.

The rectangles in the panorama images for example can be transformed into 3D positions based on the ground plane geometry. If the lower edge of the boxes represent the feet and the box does not reach the image border the 3D world position results from the intersection of a ray cast from camera center through the center of the lower box edge with the ground plane. If the foot edge of the box is not visible, the distance can be estimated by means of an average object width and the camera's projective geometry. For the depth template detector 3D positions result directly from the depth data. The laser based detectors provide the 2D position on the ground plane, which is extended into 3D by means of an average person height having a rather large variance in vertical direction.

Besides the position data, the scores from the detectors are processed into probabilities of representing a person at all, which is described in Sec. III-D.

Caused by the individual latency of the various detectors, the detections typically arrive out of order, but the output of the tracker should always represent the best estimation of the peoples' state, that is available for decision making in the application. Therefore, the tracker runs at a fixed cycle time (100ms in our case) and has an input buffer for detections that allows for sorting the observations by time stamp of the original sensor reading. If an older observation arrives, the tracker rewinds to a state before that and recomputes the whole sequence of updates with all available buffered detections. This replay strategy solves the problem of out of order detections, which alternatively is approached by a backward prediction of the latest state estimation in literature [10].

The actual tracker holds a set of hypotheses each having a Gaussian distribution for the position and velocity in world coordinates and several binary probabilities for the existence proof in each of the i sensors ($P^i(E)$). The output of the system is a subset of the position hypotheses that is filtered by a threshold for the combined existence probability $P^c(E)$.

The tracker processes the sequence of detections step by step in groups of detections belonging to the same time stamp. This is done by (i) predicting the belief state from a former time stamp to the actual observation time stamp, (ii) apply a nearest neighbor data association, and (iii) update the individual belief states using the respective detections and associations. For that update the Bayesian product of the belief distribution in the hypotheses with the distribution of the observation is used (update step of a Kalman-Filter).

B. Detectors

The images of the three wide angle cameras are projected onto cylinder coordinates ending up with a low angular resolution. Thus, these images allow person detections from close interaction ranges from $0.5\ m$ up to about $4.5\ m$. We applied and tested different detectors on these three cameras, which comprise the Deformable Part Model (DPM) [12] with its fast implementation from [13], the Fastest Pedestrian Detector in the West (FPDW) [14], and the OpenPose [3] detector. While the first two detectors are deployed on one of the two PCs of the robot, the latter one is a deep learning approach and needs special GPU hardware (one Nvidia Jetson TX2 for each camera in our case). Since RGB detectors typically have relatively high execution times and delays (0.1s FPDW, 0.2s OpenPose, 0.5s DPM) we use the laser based leg detector from [15] as supplement. This detector works on two laser range scans also covering 360° around the robot and delivers fast positional updates for tracked hypotheses. In order to perceive people early and to have a far range detector operating at distances above 10m, the robot is additionally equipped with a Kinect2 RGB-D sensor. Using a pan-tilt unit this sensor is aligned to a

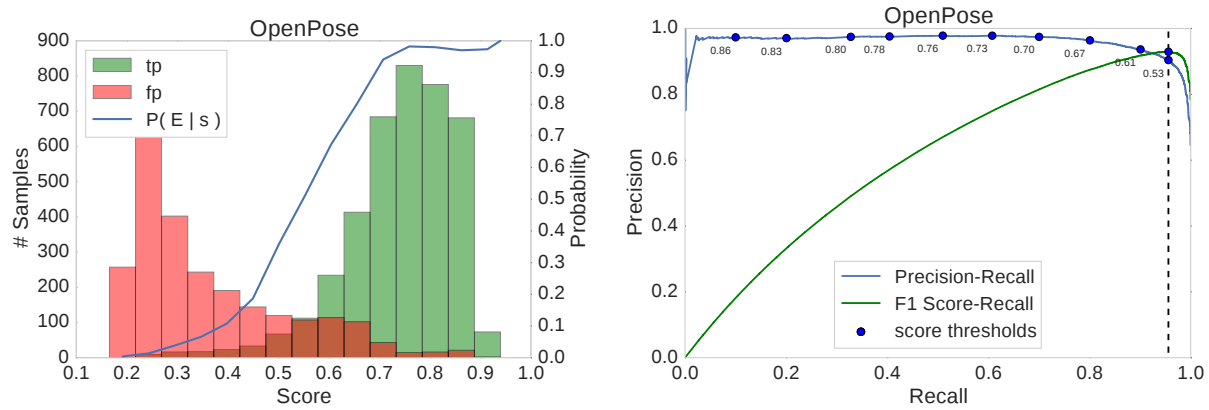

Fig. 4: Left: Binned occurrences of true positives (green) and false positives (red) in relation to their score retrieved from the classifier. The blue line is the probability, retrieved from these two distributions, that a given score with unknown class label represents a true positive detection. Right: Precision recall curve in relation to the F1 score. The optimal working point for the classifier is found with the max F1 score. For some working points on the precision recall curve (blue dots) the corresponding score threshold is given.

region which has the highest priority for our application scenarios. To the front of the robot for driving tasks and to the back when guiding people for example. With the additional depth information from this sensor, we apply the fast depth template (DT) based detector from [16].

C. Existence Probability

Once a position hypothesis H is created based on an observation in a detection module, the aim is to estimate if this hypothesis is a true positive or a false detection. To that end, in the tracker for each detector there is a module, that estimates the probability of representing a real person (existence probability) for each hypothesis. This is done per detector, since each detector has a unique update rate and an individual detection rate. These characteristics are taken into account in the process model which is used for propagating the existence belief over time. Unfortunately, typical detection approaches only yield positive detections over a certain threshold and no observations of the class non-person. Hence, a proper Bayesian integration of probabilities is only possible for the proof of existence but not for non-existence. Therefore, non-existence must be modeled otherwise. For example by not existing observations over time. Assuming that at time t a new detection D_t^i arrives for detector i, the corresponding tracker executes the prediction step (Eq. 1). There, it applies a detector specific decay τ^i, which pulls the existence probability towards 0.5 over time. 0.5 is the neutral state that does neither proof the existence nor the absence of a person.

$$\hat{P}_H^i(E_t) = 0.5 + \left(P_H^i(E_{t-\Delta t}) - 0.5\right) e^{-\Delta t/\tau^i} \quad (1)$$

Then, with each associated detection the Bayesian update is done for the hypothesis H by

$$P_H^i(E_t) = \frac{\hat{P}_H^i(E_t) P_D^i(E_t)}{\hat{P}_H^i(E_t) P_D^i(E_t) + (1 - \hat{P}_H^i(E_t))(1 - P_D^i(E_t))} \quad (2)$$

where $P_D^i(E_t)$ is the detection specific existence probability extracted from the detection score or certainty as described later.

In order to get a combined existence probability $P_H^c(E_t)$ for a hypothesis H that is supported by different detectors, the $P_H^i(E_t)$ all get multiplied together as independent probabilities.

$$P_H^c(E_t) = \frac{\prod_i P_H^i(E_t)}{\prod_i P_H^i(E_t) + \prod_i (1 - P_H^i(E_t))} \quad (3)$$

D. Mapping detection scores to probabilities

Typically, person detectors like [3, 13–16] deliver positional information in the specific sensor space along with a score indicating how certain the detector is about this position. One way to model a probability is to map the score directly to a probability. This is done by counting false positives and true positives in a histogram over the score and building the respective ratios (Eq. 4).

$$P(E|s) = \frac{\#(tp, s)}{\#(tp, s) + \#(fp, s)} \quad (4)$$

An example of such a probability function is visualized in Fig. 4 on the left for the OpenPose detection approach evaluated on our test-set from [17]. Another way to model the probability that a detection is a true positive is to determine the detectors precision. Typically, detections are just forwarded to a tracker if the scores exceed a certain threshold. One method to find such a threshold is to evaluate the detectors F1 score over a given test-set (see Fig. 4 right). The threshold with the highest F1 score gives the optimal ratio between a low amount of false detections and a high recall. Calculating the precision at this point, which is the ratio of true positives to false positives exceeding the threshold, can then directly be used as probability for all detections forwarded to the tracker. We will compare these two methods for probability determination in the experimental section where we call the

first approach *ours-dynamic-probs* and the second approach *ours-fixed-probs*.

E. Data Association

For a multi hypotheses tracker the essential step before all the updates described so far is the data association. That means which hypothesis yields which observation and thus has to be updated with it. For position trackers different approaches have been suggested in the literature. First the Euclidean distance, which is working well but ignores the variances of the hypotheses and detections. Another option is the Battachayya-coefficient as a measure for the similarity of the distributions of the hypothesis and a detection. But often this is not practical since the tracked hypothesis usually has a smaller variance than the detection, which reduces similarity also for correct matches and therefore is not comparable to the values gained by other pairs of detection and hypothesis. A further option seen in literature is the Mahalanobis distance, which takes into account the variances of the individual hypotheses in the tracker. But we found that the effect of this is counterproductive in our system. Because there is no mechanism foreseen for merging hypotheses that occur at the same position, we need a mechanism to prefer one of the competing hypotheses over the other in order to make one of them extinct. The Mahalanobis distance does exactly the opposite by reaching smaller distance values for the hypothesis with the larger variance, which usually is the one that has less support by observations over time. In order to reinforce the better supported hypothesis at one position, we developed a similarity function $s_{pos}(P(x|D), P(x|H))$ that prefers small variances in the hypotheses. $P(x|D)$ and $P(x|H)$ are Gaussian distributions over the position with mean vector μ and covariance matrix Σ.

$$s_{pos}(P(x|D), P(x|H)) = e^{-\left(\frac{|\mu_D - \mu_H| \cdot |\Sigma_H|^{\nu}}{r}\right)^2} \quad (5)$$

The parameters $r \approx 1.0m$ and $\nu \approx 0.1$ define the tolerance for position deviations and the influence of the hypothesis' variance.

Using this similarity measure, the data association is done by means of a greedy strategy. New hypotheses are generated also based on this value. Detections with a similarity below 0.5 to any of the existing hypotheses are introduced as new hypotheses. Introduction of too many hypotheses is not a problem, since before output they are filtered based on the existence probability, which only can raise above a threshold if subsequent observations (also from other sensors) will be assigned.

Hypoheses which are not supported by observations over time also need to be removed later on. Therefore, we use the fact, that uncertainties in position and existence probability grow over time if no detection takes place anymore. A threshold on $|\Sigma_H|$ is used to decide that hypotheses have to be removed from the tracking process.

IV. DATASET

Over the years, a huge amount of datasets for benchmarking detection and/or tracking approaches have been published. They include 2D laser [18–20], RGB [21] and RGB-D [22, 23] data, just to mention some of them. Unfortunately, multi sensor datasets are very rare. The only one from a robotic scenario [6] contains a static setup in the publicly available version. Moreover, datasets with ground truth labels from an external motion capture system are typically bound to a relatively small recording area. More critically, we examined ground truth failures for some frames in [6, 23] and even completely unannotated bystanders in [6].

Since our current interest in the further development of our tracking system is concentrated on the correct identification of the target person during a robotic guide procedure in populated environments, we have the need for a respective dataset. This dataset should contain multiple sensor data streams from a moving robot, while people walk behind a robot but also cross the way at all sides.

Therefore, we recorded our own data using our mobile robot platform (see Fig. 1). The dataset consists of five sequences with an overall time of $11\mathrm{min}\ 35\mathrm{sec}$, each containing a guided tour in our office building. There are five persons present in average from which one walked behind the robot in a distance of about 3m, while the others randomly walked around and crossed the guided person.

The data has been manually labeled in 3D world coordinates. Each person who is visible in at least one sensor has been annotated with their position and orientation, while the IDs are consistent if someone reappears later in a scene. This also allows the benchmarking of person re-identification on this dataset. The labels have been placed at significant key frames manually and positions and orientation were interpolated for the intermediate frames.

The images of the Fish-Eye cameras are very distorted and can not be rectified on a plane due to the large opening angle. For the application of image based detectors we therefore use a projection on a cylinder around the vertical axis. By means of that, vertical lines and proportions at one distance were preserved and only horizontal lines become curved, which is the best solution for the task of finding people with detectors trained on pinhole images. The dataset also provides access to these "rectified" images in addition to the raw images.

V. EVALUATION

To evaluate the performance of our system we use the common Multiple Object Tracking Accuracy (MOTA) metric from [24].

$$MOTA = 1 - \frac{\sum_t (miss_t + fp_t + ids_t)}{\sum_t g_t} \quad (6)$$

For every frame, we use the Hungarian Method to assign exactly one tracked hypothesis to the ground truths using a maximum valid assignment distance of $1\mathrm{m}$. Ground truths which have no assigned hypothesis count as missed (*miss*). Likewise, Hypotheses which have no assigned ground truth count as false positives (*fp*). When a hypothesis is assigned to a ground truth with another ID than in the previous frame, an ID switch (*ids*) is counted. The values of these three measures are summed up over all t frames and divided by the

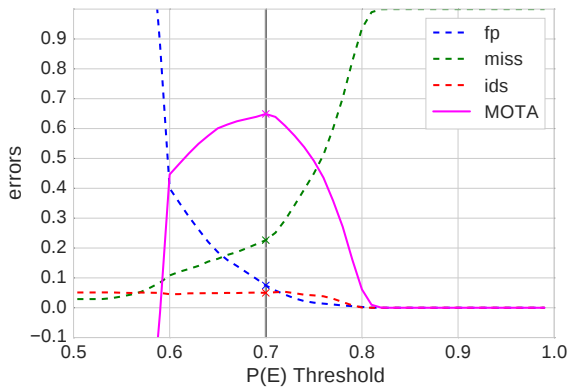

Fig. 5: Evolution of MOTA values with different existence thresholds. While higher thresholds lower the amount of false positives and ID switches, the amount of missed hypotheses increases. The threshold with the highest MOTA determines the best working point for our tracker in the application phase (vertical black line).

sum of all ground truths to make sequences with a different number of persons comparable. The resulting MOTA values can range from $-\infty$ (since the amount of false positives is theoretically unbound) to 1 for a perfect tracking of all ground truths. An example for MOTA values with different existence thresholds can be found in Fig. 5. It shows how the MOTA values change if we consider hypotheses to be certain only if their corresponding probability of existence exceeds a certain threshold. In each of the following experiments, we consider the threshold value with the highest MOTA as the working point for our tracking system. To show how our tracking method performs in comparison to other SotA approaches, we conducted several experiments on the publicly available dataset from [6] and our own dataset.

A. *Comparison on the Motion Capture Sequence*

At first, we performed experiments on the Motion Capture Sequence from [6]. This dataset consists of data from a RGB-D camera with a small field of view (FOV) in combination with a laser sensor (large FOV). Since [6] has already shown that for the best performance on this dataset these two sensor sources have to be combined, we will just provide results for a combination of detectors from these two sensor sources. With this first experiment we want to show how our approach performs in comparison to other SotA tracking approaches. To give comparable results, we used the same two detection cues (leg detector and depth template detector [15, 16]) like in [6]. Results can be found in Table I. In this experiment our approach performs on rank two. While we clearly outperform the Nearest Neighbor (NNT) and Multi Hypotheses Tracking (MHT) approach from [6], we achieve a slightly better (5% MOTA) performance than the Multi Modal Person Tracker (MMPT) from [10]. The extended NNT approach on the other hand, with a track initialization logic where just moving hypotheses are considered as certain, outperforms our approach with about 11% MOTA. However, since the extended NNT approach is not able track persons standing still, and most of this dataset consist of moving persons, we consider our approach to be the best tracking approach for general person appearances.

Approach	MOTA	ids	fp	miss
[6] NNT*	18.1%	52	77.6%	3.7%
[6] MHT*	17.8%	76	77.7%	3.6%
[6] Extended NNT*	77.4%	62	16.5%	5.4%
[10] MMPT	61.0%	158	11.0%	26.5%
ours-fixed-probs	66.6%	59	1.7%	31.9%

TABLE I: Tracking performance on the Motion Capture Sequence from [6]. All approaches use the same detection cues [15, 16] as input. Results for approaches marked with * are taken from the corresponding publication.

Approach	Detectors	MOTA	ids	fp	miss
[10] MMPT	[15] [13]	34.8%	193	22.6%	40.6%
ours-fixed-probs		60.7%	167	12.0%	24.0%
[10] MMPT	[15] [14]	29.9%	183	25.5%	42.8%
ours-fixed-probs		48.4%	158	11.1%	37.3%
[10] MMPT	[15] [3]	35.9%	334	28.4%	32.3%
ours-fixed-probs		64.9%	246	7.5%	22.6%
[10] MMPT	[3, 13–16]	47.2%	141	37.2%	14.2%
ours-fixed-probs		68.7%	104	9.3%	19.9%

TABLE II: Tracking performance on the Motion Capture Sequence from [6] with different detection cues: Leg detector [15], DPM [13], FPDW [14], OpenPose [3], DT [16].

Approach	Detectors	MOTA	ids	fp	miss
[10] MMPT	[15] [13]	60.3%	110	0.7%	38.5%
ours-fixed-probs		69.9%	172	7.0%	21.9%
ours-dynamic-probs		69.8%	189	6.3%	22.5%
[10] MMPT	[15] [14]	52.7%	71	0.1%	46.9%
ours-fixed-probs		64.3%	135	11.8%	22.9%
ours-dynamic-probs		64.5%	113	11.3%	23.2%
[10] MMPT	[15] [3]	52.9%	109	1.4%	45.3%
ours-fixed-probs		64.0%	200	10.2%	24.3%
ours-dynamic-probs		64.7%	185	11.2%	22.7%
[10] MMPT	[3, 13–16]	64.3%	113	6.1%	29.1%
ours-fixed-probs		68.1%	166	9.4%	21.2%
ours-dynamic-probs		68.7%	209	8.4%	21.2%

TABLE III: Tracking performance on our Guide Sequences with different with different detection cues: Leg detector [15], DPM [13], FPDW [14], OpenPose [3], DT [16].

With the second experiment we want to show how our approach performs with different combinations of detection cues as input. Results can be found in Table II. It can be seen that detection cues from sole RGB data in combination with a leg detector perform generally worse than a leg detector in combination with the depth template based detector [16] from the first experimental series. This is contradictory to previous experiments we performed for the pure detection task [17]. There, we have shown that every RGB detector outperforms the simple depth template based detection approach [16] in the image space. We analyzed this phenomenon and figured out that the 2D detection quality is not the only quality feature for a detector when it comes to tracking in world coordinates. While the detections in image space seem quite well, the projection into 3D space

leads to offsets which makes the data association harder for detectors without precise depth information. However, our approach is more suitable to deal with this issue than the competitor from [10]. While our approach was just 5% better using the DT detector [16] as second detection cue, this gap rises by at least 20% when using RGB detectors instead. This comes from the track initialization logic they applied in [10]. There, just a single false detection or erroneous data association leads to a certain hypothesis even when just the leg detector offers continuous support. However, covering the complete surroundings of the robot with small FOV and power consuming RGB-D cameras, like the Kinect2, would be impossible for mobile platforms with restricted resources. Therefore, we now turn to our dataset with a sensor setup suitable for a mobile platform. The results in Table III show that our approach for multi sensor tracking also performs well on a dataset with an omni directional RGB camera setup in addition to two large FOV laser sensors. The method to determine the existence probability seems to have no significant influence on the tracking quality. This on the one hand can be explained with the highly differing environment (supermarket) of the dataset on which we determined the detection probabilities. On the other hand, our test dataset shows only few situations suffering from a large amount of false positive detections where the dynamic-probs approach can show its supremacy. However, duly determining existence probabilities as well as to fuse them in a probabilistic manner gives a great boost for the tracking quality compared to the approach from [10] for all detector combinations.

At last, we examined the tracking quality for different ranges and sensor areas. For this experiment, the configuration using all available detectors in combination was applied. The leg detector [15] on both lasers. DPM [13], FPDW [14] and OpenPose [3] on the Fish-eye cameras. DPM [13], FPDW [14], OpenPose [3] and the depth template on the Kinect2. Results can be found in Fig. 6. There, the MOTA values are given for a tracking range up to $5\mathrm{m}$, $10\mathrm{m}$ and $20\mathrm{m}$ as well as for the sensor area covered by all sensors and the one just covered by the laser scanners and fisheye cameras. It can be seen that the tracking is more accurate in close proximity to the robot, which is reasonable because of the low resolution of our wide angle cameras and the resulting shorter detection ranges. For the area, which is additionally covered by the Kinect2, the MOTA values rise by 5% for longer distances because of the high resolution RGB and depth images. This is important for applications where a specific area has a higher importance, like in our guiding scenario where the person of interest follows the robot in the viewing area of the Kinect2.

To make a statement about the real-time capability of our system, we measured the runtime on the robot's i7-7700T CPU using just a single core. While benchmarking on our dataset, each tracking cycle took $1.6\mathrm{ms}$ on average. This enables a maximum frame rate of $625 fps$. Since our fastest detector (laser based leg detector) runs at $15 fps$, we are able to process results from all detectors of our system immediately. However, complex real world applications [1, 2] require

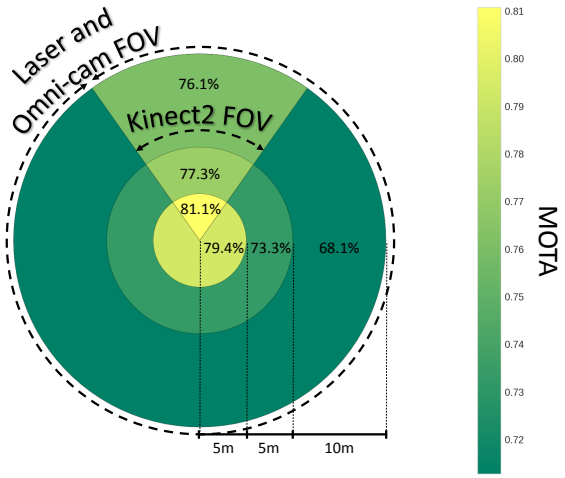

Fig. 6: MOTA values of different evaluation distances and angles in relation to the robot coordinate system.

various other robotic skills in combination with tracking, like navigation, scene understanding, and application services. Therefore, we usually let the tracker operate at $10 fps$ only. This frequency is high enough for our applications and only utilizes 1.6% of one CPU core, which leaves enough computation capacity for other tasks and robotic services.

VI. CONCLUSION & FUTURE WORK

We have presented a filter based tracking system which takes probabilities retrieved from several SotA detectors into account to distinguish true hypotheses from false ones. The tracking accuracy using the MOTA criterion exceeds the values of several competitors on a publicly available dataset. Moreover, we introduced a novel dataset for robotic tracker evaluation which is, for the best of our knowledge, the first one which comprises multi sensor data from more than two sensors and was recorded in a highly dynamic scene on a moving robot. One obvious fact, which is depicted in the tables of Sec. V, is that our tracking approach prefers a high miss rate in favor of a low false positive rate when the existence threshold is optimized to maximize the MOTA criterion. This might be an issue in scenarios where a high person perception rate is more important than the low number of false hypotheses. A static robot for example, which observes persons passing by to find individuals who are willing to interact, like in our project FRAME[3], might suffer from this optimization. However, the working point of existence probability threshold in the hypothesis filter can be varied freely during runtime. It might be even possible to adapt the threshold dynamically depending on the actual requirements in various application states.

REFERENCES

[1] H.-M. Gross, S. Meyer, R. Stricker, A. Scheidig, M. Eisenbach, St. Mueller, Th. Q. Trinh, T. Wengefeld, A. Bley, Ch. Martin, and Ch. Fricke, "Mobile Robot Companion for Walking Training of

[3]http://www.frame-projekt.de/

Stroke Patients in Clinical Post-stroke Rehabilitation," in *International Conference on Robotics and Automation (ICRA)*, 2017, pp. 1028–1035.

[2] H.-M. Gross, St. Mueller, Ch. Schroeter, M. Volkhardt, A. Scheidig, K. Debes, K. Richter, and N. Doering, "Robot Companion for Domestic Health Assistance: Implementation, Test and Case Study under Everyday Conditions in Private Apartments," in *International Conference on Intelligent Robots and Systems (IROS)*, 2015, pp. 5992–5999.

[3] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[4] J. Redmon and A. Farhadi, "Yolov3: An incremental improvement," *arXiv*, 2018.

[5] W. Luo, J. Xing, A. Milan, X. Zhang, W. Liu, X. Zhao, and T.-K. Kim, "Multiple object tracking: A literature review," *arXiv preprint arXiv:1409.7618*, 2014.

[6] T. Linder, S. Breuers, B. Leibe, and K. O. Arras, "On multi-modal people tracking from mobile platforms in very crowded and dynamic environments," in *International Conference on Robotics and Automation (ICRA)*, 2016, pp. 5512–5519.

[7] K. O. Arras, S. Grzonka, M. Luber, and W. Burgard, "Efficient people tracking in laser range data using a multi-hypothesis leg-tracker with adaptive occlusion probabilities," in *International Conference on Robotics and Automation (ICRA)*, 2008, pp. 1710–1715.

[8] C. Dondrup, N. Bellotto, F. Jovan, and M. Hanheide, "Real-time multisensor people tracking for human-robot spatial interaction," in *Workshop on Machine Learning for Social Robotics at International Conference on Robotics and Automation (ICRA)*. ICRA/IEEE, 2015.

[9] N. Bellotto and H. Hu, "Computationally efficient solutions for tracking people with a mobile robot: an experimental evaluation of bayesian filters," *Autonomous Robots*, vol. 28, no. 4, pp. 425–438, 2010.

[10] M. Volkhardt, Ch. Weinrich, and H.-M. Gross, "People tracking on a mobile companion robot," in *International Conference on Systems, Man, and Cybernetics (SMC)*, 2013, pp. 4354–4359.

[11] T. Linder, F. Girrbach, and K. O. Arras, "Towards a robust people tracking framework for service robots in crowded, dynamic environments," in *Assistance and Service Robotics Workshop (ASROB-15) at the International Conference on Intelligent Robots and Systems (IROS)*, 2015.

[12] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *transactions on pattern analysis and machine intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[13] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *European Conference on Computer Vision (ECCV)*. Springer, 2012, pp. 301–311.

[14] P. Dollár, S. Belongie, and P. Perona, "The fastest pedestrian detector in the west," in *British Machine Vision Conference (BMVC)*, 2010.

[15] K. O. Arras, Óscar Martínez Mozos, and W. Burgard, "Using boosted features for the detection of people in 2d range data," in *International Conference on Robotics and Automation (ICRA)*, 2007.

[16] D. Mitzel and B. Leibe, "Close-range human detection for head-mounted cameras," in *British Machine Vision Conference (BMVC)*, 2012.

[17] B. Lewandowski, J. Liebner, T. Wengefeld, S. Müller, and H.-M. Gross, "A fast and robust 3d person detector and posture estimator for mobile robotic applications," in *will be published on International Conference on Robotics and Automation (ICRA)*. IEEE, 2019.

[18] L. Spinello and R. Siegwart, "Human detection using multimodal and multidimensional features," in *International Conference on Robotics and Automation (ICRA)*, 2008, pp. 3264–3269.

[19] C. lvarez Aparicio, . M. Guerrero-Higueras, M. C. C. Olivera, F. J. Rodrguez-Lera, F. Martn, and V. Matelln, "Benchmark dataset for evaluation of range-based people tracker classifiers in mobile robots," *Frontiers in Neurorobotics*, vol. 11, p. 72, 2018.

[20] Ch. Weinrich, T. Wengefeld, Ch. Schroeter, and H.-M. Gross, "People detection and distinction of their walking aids in 2D laser range data based on generic distance-invariant features," in *International Conference on Robot and Human Interactive Communication (RO-MAN)*. IEEE, 2014, pp. 767–773.

[21] A. Ess, B. Leibe, K. Schindler, , and L. van Gool, "A mobile vision system for robust multi-person tracking," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2008.

[22] M. Luber, L. Spinello, and K. O. Arras, "People tracking in rgb-d data with on-line boosted target models," in *International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 3844–3849.

[23] M. Munaro, F. Basso, and E. Menegatti, "Tracking people within groups with rgb-d data," in *International Conference on Intelligent Robots and Systems (IROS)*, 2012.

[24] K. Bernardin, A. Elbs, and R. Stiefelhagen, "Multiple object tracking performance metrics and evaluation in a smart room environment," in *Sixth IEEE International Workshop on Visual Surveillance, in conjunction with ECCV*, vol. 90. Citeseer, 2006, p. 91.