2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Mapping of Passive UHF RFID Tags with a Mobile Robot using Outlier Detection and Negative Information

Artur Koch and Andreas Zell

***Abstract*— In this paper we propose a novel approach to classify detection events from a stream of radio-frequency identification (RFID) measurements for the purpose of mapping RFID transponders. Since raw readings from RFID readers only provide information on positive read attempts, i.e. the detections of a tag, we propose an outlier filter method solely based on the spatial extent of the sensor model that is used for the mapping process. Furthermore, we use this filter to actually classify detections as well as non-detections of tags into valid and invalid positive as well as negative detection events. We incorporate the different classes into our mapping pipeline and introduce several extensions to improve the mapping accuracy. Experimental results including the classification and mapping accuracy are presented to prove the effectiveness of our approach.**

I. Introduction

Radio-frequency identification (RFID) based mapping and localization approaches have become a vital research topic in the literature in recent years. Featuring low costs, easy deployment and contactless identification through unique identifiers, RFID tags are already present in many different application scenarios, like warehouses, supermarkets or logistics. In the context of this work, we focus on the RFID-based mapping task of passive *ultra-high frequency* (UHF) RFID tags that are spread out in the surroundings and detected by an RFID reader mounted on a mobile robot traversing the environment. By utilizing our autonomous mobile system (see Fig. 1) we are able to carry out the mapping task of the stationary transponders in parallel to the often necessary inventorying process of goods, given an RFID system and the pose of the mobile agent.

Since traditional RFID systems only report the observations of tags through streams of tag Ids, the respective number of detections and *received signal strength* (RSS), most mapping approaches rely on Bayesian inference based on sensor models. We follow this paradigm, but as opposed to other related work, focus on the classification of the observed readings to filter out false positive read attempts, or *ghost-detections* as they are also referred to in the literature. In particular, we use the bounding box of the underlying 2D/3D sensor model to train a filter on positive observations and remove ghost-detections, which generally are the result of environmental effects (like reflection or diffraction) on the electromagnetic waves of the radio-frequency signal. After having trained on the positive observations, we are able to utilize the filter as a classifier to discriminate between true and false negative observations, i.e. non-detections, of the tags for the mapping process.

A. Koch is with the the Chair of Cognitive Systems, headed by Prof. A. Zell, Computer Science Department, University of Tübingen, Sand 1, D-72076 Tübingen, Germany {artur.koch, andreas.zell}@uni-tuebingen.de

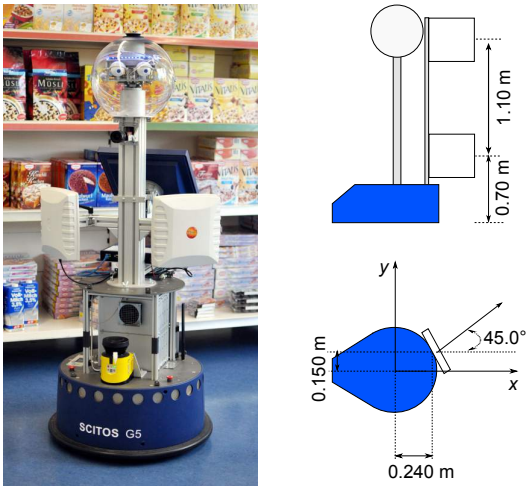


Fig. 1. Scitos G5 service robot equipped with a UHF RFID system.

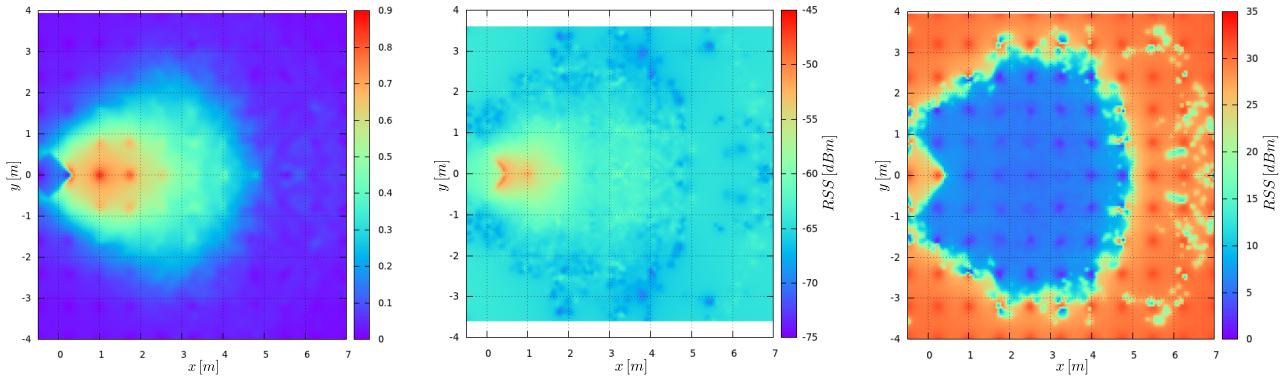
Additionally, since non-detections usually dominate the detection event set of single tags, we use the per tag information on first and last detection timestamps reported by the reader during a predefined inquiry period, to interpolate and thus generate additional positive detections. On the one hand, this gives us a higher resolution in the stream of positive detections and thus a better balance regarding negative events. On the other hand, this also leads to a bias of the estimated posterior, which makes it necessary to utilize a redundancy or similarity filter on observations to enforce a certain spread in the 3D/6D-domain of the detection event poses.

The remainder of the paper is structured as follows. In Sect. II we survey related works. Sect. III gives an overview over Bayesian inference for mapping. In Sect. IV we introduce our classification approach for RFID observations. Thereafter, we provide details on the interpolation and filtering process in Sect. V. Experimental results are presented in Sect. VI. Finally, in Sect. VII we draw conclusions.

II. Related Work

The literature on related work is broad, in the following we give a brief overview over most related publications.

Hähnel *et al.* [1] were the first to introduce particle filters for Bayesian inference in the context of UHF RFID based mapping. They utilized the robot positions computed by a laser-based SLAM (simultaneous localization and mapping) approach to estimate RFID transponder positions with a



(a) (b) (c)

Fig. 2. Sensor models (quartic kernel regression): 2D detection rate sensor model (a) and 3D RSS sensor model with mean (b) and standard deviation (c) at $z = 0$. The antenna is located at $(0, 0)$ and parallel to the y-axis.

RFID sensor model. In a further stage they used the generated RFID map for Monte Carlo self-localization solely based on RFID and odometry in an indoor environment. Vorst *et al.* [2] were able to extend the approach and learn sensor models in a semi-autonomous fashion based on data gathered from exploring an environment with a mobile robot.

Deyle *et al.* [3] employed a sensor model characterizing antenna properties and multi-path propagation to estimate tag positions via particle filtering using six antennas. Liu *et al.* [4] were able to detect transponder movement by using a Markov localization variant for tagged objects with RFID.

Various approaches have been proposed to fuse sensors or to improve particle filter based state estimation. Deyle *et al.* [5] aligned 2D RSS distribution images with RGB and 3D laser data to localize objects in 3D. Liu and West [6] introduced a kernel and auxiliary particle filter based technique for resampling without loss of prior information. Fox *et al.* introduced an alternative approach to reduce the sample size by means of KL-distance to improve particle filter performance in [7].

Joho *et al.* [8] introduced RSS in combination with detection probability models to estimate tag positions. We follow their approach and utilize the recent 3D extensions by Liu and the authors in [9] to estimate tag positions in 3D and to evaluate the performance of the herein proposed methods. To our knowledge, there have not yet been any relevant publications related to the actual classification of RFID data.

III. Sensor Model based Mapping Extensions

Bayesian inference based mapping is a well studied topic in the robotics community, we therefore do not go into the basics and refer the interested reader to the literature, e.g. [10], [1] and [8]. In the following we briefly describe our recent changes to the mapping module from our previous approach in [9]. In contrast to our previous work, we utilize a two-dimensional *detection rate sensor model* (DRM). This is due to the fact that the 3D version seems to underestimate the likelihood especially in the outer regions of the model, which is related to the bigger span (i.e. additional dimension) and thus a lower sample density throughout the regression. Moreover, the detection rate model is trained on a binary feedback to estimate the detection probability, whereas the *RSS model* (RSSM) gets quantitative values for each sample to compute its mean and variance values. This makes it necessary to have a higher support density in the case of the DRM as opposed to the RSSM, which seems to deliver a good approximation from less samples for the 3D case.

To get smoother models we extend the 3D part of our regression to use quartic kernels (see Fig. 2) for the mapping process as introduced by Vorst *et al.* in [2]. Additionally, we now initialize the particles of the particle filter uniformly in a 3D volume around the first observation instead of using regularly sampled positions (cf. [9]).

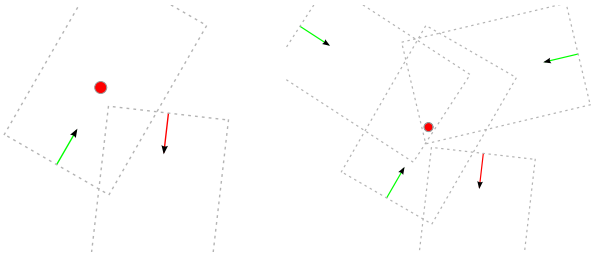
IV. Outlier Detection and Classification

The major motivation behind the outlier detection task is to avoid a corruption or degeneration of the posterior estimate through *false positive* (*fp*) detections. We regard these ghost-detections mostly as reflected electromagnetic waves that lead to invalid detections of a tag at a robot, or more precisely, antenna pose, where the tag is impossible to be detected. Therefore, it is obvious that those detections, if treated as *true positive* (*tp*) observations in the mapping process, will lead to a degeneration of the posterior. Regarding real-world data from our test environment, which features metallic radiators attached to one wall, we identified up to 18.8% ghost-detections inside the data streams of single tags.

A. Positive Detections

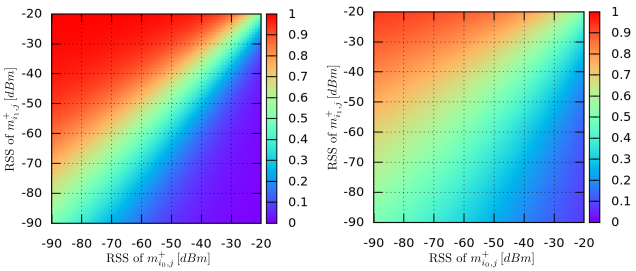
The idea of the approach is first based on the fact, that the RFID reader is guaranteed to only report the unique Ids of the stationary tags that actually have been detected, and thus are present in the read range of the device. Additionally, we assume that the underlying sensor model, with a high confidence, approximates the major read range of the active RFID system. Since the reader never reports a positive detection of a tag that is not present in the environment, we assume the positive detections to be unambiguous and mutually exclusive regarding negative events.

Given those assumptions, we build a voting-based classification method that uses the 2D/3D *oriented bounding box* (OBB) of the 2D/3D sensor model to identify conflicting



(a) (b)

Fig. 3. Outlier detection: (a) Antenna poses (represented as arrows in 2D) of conflicting ghost-detection (red) and valid observation (green) of a tag. (b) Multiple valid observations are used to identify the ghost-detection. The position of the tag (red circle) is presented for visualization purposes only and is *not* known in the outlier detection task.



(a) (b)

Fig. 4. Penalty function example for β = -18 (a) and β = -8 (b). The penalty value of the positive detection i_0 of tag j depends on its own RSS value (x-axis) as well as the signal strength of the other event i_1 (y-axis).

detection events for each tag (see example in Fig. 3). We neither imply any knowledge about the environment nor the actual position of the tag, since this is the central task of the background mapping process.

In our case, a pair of positive detection events $m^+_{i_0,j}$ and $m^+_{i_1,j}$ of tag j is (with a high confidence) considered to be conflicting, if the corresponding bounding volumes of the sensor model *do not intersect*. As a result, we attach a penalty score from an arbitrarily chosen penalty function

$$f_p(m^+_{i_0,j}, m^+_{i_1,j}) \tag{1}$$

to such pairs and repeat the process for all pairs of detection events of tag j. If the data stream provides an appropriate *signal-to-noise ratio* (SNR) regarding true and false positive detections, and the punishment function is chosen accordingly, we should succeed to discriminate between *tp* and *fp* readings based on their penalty scores.

The removal (or classification) of false positives is carried out in an iterative fashion. After having updated the penalty scores of all participating events, we remove the detection with the maximum penalty score and decrement the scores of the events previously punished by this candidate. We repeat this until none of the events features a penalty score, or until a predefined penalty threshold τ is reached. All remaining events are considered to be valid positive events while the removed events are marked as false positives.

B. Penalty Function

Obviously the choice of the penalty function (1) has a great impact on the success rate of the proposed method. We design our function to account for different characteristics of the observations like the maximum RSS inside a pair and the respective RSS difference. Since the amount and spread of *tp* events inside the data stream will make up for the lack of complexity, a naive punishment function would just uniformly increase the penalty scores of both detection events. We extend this function by utilizing RSS information. Based on a simple relation of RSS to distance, we assume detections with higher RSS values to be closer to the tag, and thus be more reliable. Although this assumption does not always hold (e.g. regarding orientation, environmental effects), we try to avoid relying on complex models, since the complexity of our method is quadratic in regard to the number N of positive detections ($\frac{N(N-1)}{2}$ pair-wise intersection tests and penalty updates necessary).

Our proposed function for the penalty increment of detection $m^+_{i_0,j}$ when compared to detection $m^+_{i_1,j}$ is derived from the Fermi-Dirac statistics and defined as follows:

$$f_p(m^+_{i_0,j}, m^+_{i_1,j}) = \frac{1}{\exp\left(\beta\left(\frac{r_{i_0,j}}{r_{i_0,j}+r_{i_1,j}} - 0.5\right)\right) + 1} \tag{2}$$

with $r_{i,j}$ giving the RSS of the i-th detection event of tag j.

Using the RSS values of single detections and their differences, we enforce a higher penalty for the observation with the smaller RSS. Additionally, since we assume a Gaussian RSS error distribution inside our observations, small differences in RSS values do not get a significantly imbalanced treatment. Fig. 4 shows two examples of the function for different values of β, which may be used to scale the balance bandwidth of the penalty function to fit the variance range.

C. Negative Detections

Using positive events, we were able to distinguish true and false positive events on conflicting model boundary intervals. After having removed *fp* candidates, the classifier is actually already trained on positive events and may be applied to negative events in a straightforward fashion. The penalty function treats negative events implicitly correct if the RSS is assumed to be zero ($-\infty$ in dBm) and thus punishes all non-intersecting negative observations. As a result, non-conflicting negative events represent the *false negative* (*fn*) detection events, since the tag should have been, but was not detected at this observation's pose. Vice versa, conflicting negative events form the *true negatives* (*tn*) representing observations where the tag was obviously outside the detection range of the RFID system.

V. Observation Interpolation and Filtering

A. Observation Interpolation

In our work, the RFID system is configured to deliver RFID readings at a certain frequency f_0 and thus buffers the readings obtained in $T_0 = \frac{1}{f_0} \approx 0.5\,s$ in hardware. This is done to minimize network bandwidth overhead on the mobile systems. As a result, we get aggregated tag detection information in the RFID data reported by the reader, i.e. a tag may be detected multiple times throughout T_0 by the

same antenna but is only reported once with the aggregated number of detections. Fortunately, the reader also reports so-called first- and last-seen timestamps for each tag. Utilizing this information, we apply a simple linear interpolation based on the start and end antenna poses to generate isolated observations for each cluster of detections of a single tag inside the raw RFID data. This is valid since the maximum translation and rotation velocities (usually $v_t^{max} = 0.2\frac{m}{s}$ and $v_r^{max} = 0.7\frac{rad}{s}$) as well as T_0 are constrained throughout our experiments. Through interpolation we get a higher resolution regarding observation poses as well as a better balance regarding non-detections, which usually dominate the observation set.

B. Observation Filtering

To avoid a degeneration of the posterior estimate due to consecutive similar observations, which may also be interpreted as an "overfit" of the estimate towards this observation, we apply a redundancy filter. This problem also arises in other robotics related applications (e.g. laser based localization and mapping), and is solved by enforcing constraints on the observation streams. In our application, for example, successive similar observations of a tag at the same antenna pose would lead to a convergence of the tag's position estimate towards the maximum of the likelihood function approximated by the sensor model in use.

Since our observation stream features a strict chronological ordering, we evaluate each new observation based on the previous valid detection using different predefined threshold constraints: euclidean and angular distance as well as RSS and time difference. If the observation in question complies with any of the supplied constraints, we rate it as useful regarding information gain, else we ignore the detection. As a result we enforce a minimal spread of consecutive observations without restricting the positive set too much and thus losing required information. This also has a positive balancing effect on the global distribution of observations in the target domain.

VI. Experimental Results

We carried out various experiments on real-world data to validate effectiveness and accuracy of the proposed methods. In order to get comparable results, we focused on our experimental data sets[1] provided in [9]. We only used the data of the wide baseline configuration (with two antennas vertically aligned, see Fig. 1 and cf. [9]), since those achieved peak accuracies in our previous approach. The data set consists of 7 log files, with overall 16,773 buffered RFID readings (693,521 positive observations) from a trajectory with a length of around $1.5\,km$ in approx. 143 minutes.

The general pipeline stages are as follows: (1) interpolate observations, (2) classify observations, (3) filter redundant observations and finally (4) apply mapping using the remaining set of observations. Since the mapping approach is independent of the previous stages, we enable or disable the single data preprocessing steps for different setups to obtain results for comparison. Generally, if the interpolation is enabled, the number of positive observations rises by a factor of two to 1,425,240 as compared to 693,521 before.

[1] http://www.cogsys.cs.uni-tuebingen.de/datasets/iros2013

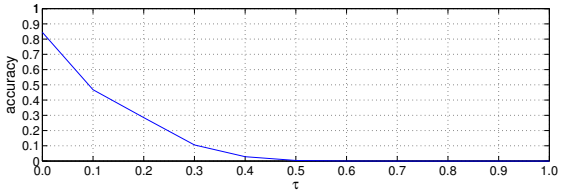


Fig. 6. Impact of penalty threshold τ on the classification accuracy of ghost-detections (false positives). τ is selected as a percentage of the positive events supplied to the classifier (x-axis). The accuracy rates were computed as a weighted mean over all tags.

A. Mapping

To not introduce influencing side-effects and keep parameters inside the particle filter fixed, we turn off resampling as well as KLD-sampling [7] or perturbation [6] throughout all experiments. To still ensure a satisfying resolution even in 3D, we set and keep the number of particles fixed at 100,000 throughout this series of experiments. We initialize the particles uniformly around the first observation pose as a cylinder with a radius of $7\,m$ (the maximum detection range) and a height of $3\,m$ (approx. distance of floor to ceiling). For statistical conformity we cross-validate over all tags. This is done by building a model for the current tag using the RFID measurements of all other tags in the log files and is derived from our application scenario, where we wish to estimate the positions of unknown tags by taking a previously generated model for the sensor. Each experiment is repeated at least three times (estimations with the mapping extension are running almost every time to track effects of the uniform random particle filter initialization), thus giving us at least 213 estimation samples over all 71 tags.

First of all, we want to examine the effects of the extensions to the mapping module (see Sect. III). Therefore we compute 2D position estimations for the 71 tags mapped in our previous work using all positive observations. The mean abs. mapping error for all tags drops to $0.212\,m$ featuring an improvement of approx. $2.8\,cm$ (i.e. around 10%) as compared to our previous work, which had a mean abs. mapping error of $0.24\,m$.

B. Classification

We tested our classifier on the data sets. Unfortunately it is almost impossible to generate real ground truth for the classes, as we would need to keep track of each detection during the inquiry process of the RFID system and label them accordingly. To get a good estimation for comparison, we did OBB overlap tests of the sensor model boundary of the observation to be classified with the true position of the detected tag, which again is unknown to the classifier itself.

Fig. 6 shows the impact of the penalty threshold τ on the classification accuracy of ghost-detections. The results indicate, that higher values of τ reduce the accuracy on *fn* observations while leading to slightly higher *tp* observation

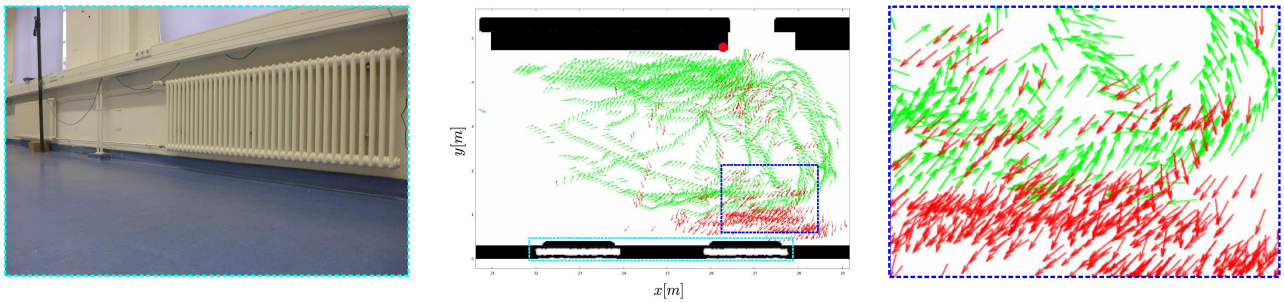


Fig. 5. Radiators in our test environment (left, light blue) and classification result example of tag with 18.8% ghost-detections (middle, right): From overall 4420 observation poses (represented as arrows) the classifier detected 3619 true positive (green) and 801 false positve (red) observations. Compared to our generated ground truth, 99.77% of true positives were classified correctly, while 95.2% of ghost-detections were correctly found and removed. The true tag position (red circle) is visualized for demonstration only and is not known to the classifier.

accuracies (always over 99.95%). Obviously, a higher threshold keeps more false events and may be used to specify the belief into the classifier, or more precise, into the model of the sensor used for classification. It may be useful if the approximation of the sensor model is not well in tune with the RFID system.

Different values of β did not have a significant influence on the classification accuracy throughout our experiments. This may be due to the fact that the assumption of a simple relationship between RSS and distance mostly does not hold and thus does not introduce a significantly better performance of the classifier. On the one hand it is obvious, that the RF signal propagation is much more complex and the implied relation does not give a close approximation. On the other hand it is surprising, since it shows that it is sufficient to punish both detection events equally to get good results, which reduces the complexity of the classification. Nevertheless, if the application needs to get rid of as much noise as possible, the proposed function slightly improves the *fp* accuracy depending on the choice of τ as can be seen in Tables I, II and III.

We set τ to 0 and β to -18 as a conservative trade-off for *tp* and *fn* observation accuracy and compute weighted mean accuracies for all classes. Fig. 5 shows an example of the classification result for a tag with 18.8% ghost-detections. As can be seen, the classifier is able to detect a majority of ghost-detections, mostly pointing towards the radiators inside our test environment. The weighted mean accuracy of correctly identified *tp* events for all tags in our log files is at 99.9% (see confusion matrix Table I), proving that the classifier does not remove a significant amount of valid observations. In the case of ghost-detections, we get a mean accuracy of 84.3% for all examined tags. Applying the classification to *tn* and *fn* events delivers us 93% and 95% respectively. To further examine the value of the classifier, we supply the different classes to the mapping module.

We remove the ghost-detections and repeat the experiment of Sect. VI-A. This improves our mean mapping accuracy by approximately $0.5\,cm$ to a mean abs. localization error of $20.7\,cm$. The small improvement is mostly due to the fact, that many ghost-detections do not influence the importance weights of the discrete representation of the posterior likelihood, since they are out of the particle filter range.

TABLE I

CONFUSION MATRICES OVER ALL TAGS FOR $\tau = 0$ AND $\beta = -18$. ROWS REPRESENT GROUND TRUTH BASED ON TRUE TAG POSITION, COLUMNS REPRESENT RESULTS OBTAINED FROM THE CLASSIFIER.

positive	*tp*	*fp*
tp	209127	104
fp	1589	8561

negative	*tn*	*fn*
tn	1541005	117217
fn	12456	491707

TABLE II

CONFUSION MATRICES FOR $\tau = 0.3$ AND $\beta = -18$

positive	*tp*	*fp*
tp	209231	0
fp	9836	314

negative	*tn*	*fn*
tn	837706	820516
fn	0	504163

TABLE III

CONFUSION MATRICES FOR $\tau = 0.3$ AND $\beta = -4$

positive	*tp*	*fp*
tp	209231	0
fp	10058	92

negative	*tn*	*fn*
tn	837858	820364
fn	0	504163

Other applications dealing with RFID might experience a bigger profit if ghost-detections exhibit a high influence on the application's outcomes.

C. Utilizing Negative Information

To further utilize the results of the classification, we evaluate the influence of the two negative classes on the mapping accuracy. In this case we integrate false negative observations with the inverse DRM (cf. [9]). Additionally, we treat true negative observations as an information criterion to remove particles that overlap with the *tn* observation's sensor model boundaries from the particle filter. Regarding Bayesian inference, this is similar to applying a sensor model with a 0-likelihood inside the model and unknown anywhere else, and thus a valid approach since we assume those observations to be true negative.

By adjusting the threshold parameter τ for the identification of *tn* we influence the amount of observations considered to be valid true negatives, and therefore express our belief into the classifier's *tn* accuracy. If we set τ too low, the particles inside the particle filter might completely vanish, since invalid true negatives (i.e. false negatives) get treated incorrectly. This would remove particles that are supposed to

TABLE IV

OVERVIEW OVER SINGLE EXTENSIONS AND AGGREGATE RESULTS

	error (cm)	error (%)	particles (%)
Previous [9]	24.0	100	100
+ Mapping Extensions	21.2	88.3	100
+ False Positive Elimination	20.7	86.3	100
+ Negative Information	20.7	86.3	≈ 25
+ Interpolation and Filtering	20.2	84.1	≈ 25

be participating in the estimation process and thus degenerate or corrupt our posterior estimate. Taking into account our previously estimated classifier accuracy, we set the threshold parameter for true negative events to the 20% quantile of the penalty score distributions inside each tag's classifier throughout this series of experiments. Thus all negative observations with a penalty score above the 20% quantile are identified as outliers (i.e. as true negatives) and used to remove particles during the estimation process.

The results show firstly, that around 75% of all particles get removed throughout our experiments while preserving the particle density inside the major regions of interest. Additionally, per tag computation times are reduced significantly by approx. a factor of four to eight depending on the amount of detections. In our experiments, the straightforward integration of false negatives with the inverse sensor model unfortunately does not deliver significantly better results and is even very costly regarding performance since *fn* still outnumber *tp* events by far.

D. Interpolation and Redundancy Filtering

In our final experiment we enable interpolation and/or filtering of redundant events. Throughout our experiments we observe that only enabling interpolation of positive events generally has a negative effect on the mapping accuracy, the errors mostly rise by $1 - 2\,cm$, while enabling only the filtering step slightly improves the results. Best results are achieved through the combination of both with the previous extensions, where the mean absolute localization error again drops slightly to $20.2\,cm$ as compared to the previous error of $20.7\,cm$. Although the additional positive events would give us a higher load regarding computations, through filtering out redundant events the performance even improves slightly.

Table IV summarizes the different improvements through the single stages of the pipeline and shows the aggregated results of all previous stages combined.

VII. CONCLUSIONS AND FUTURE WORK

In this paper we presented a classifier for the detection of true and false positive as well as negative RFID measurements. The classifier may be adjusted to the user's choice of preference in respect to accuracy and confidence by adjusting parameters of the sensor model range, the threshold τ and the bandwidth β. In our experiments, the classifier was able to correctly identify 99.9% true positive, 84.3% false positive as well as 93% and 95% true and false negative events, respectively.

We evaluated the classifier in the context of mapping RFID tags. Through our extensions we were able to successfully prune false positive events. By using the information gain from the classifier and a convenient choice of the threshold parameter τ we furthermore improved the performance of the mapping application without any loss in mapping accuracy. Combined with the performance gain from observation filtering our approach is suited for the real-time application on our mobile platforms. The final results obtained through our proposed methods feature an overall mean abs. mapping error improvement of $3.8\,cm$ from before $24\,cm$ to $20.2\,cm$.

For future work, first, we would like to test the classifier in other RFID related applications. Moreover, since we did not succeed to efficiently utilize false negatives, further evaluation should be promising in regard to negative information integration (e.g. [11], [12]).

REFERENCES

[1] D. Hähnel, W. Burgard, D. Fox, K. P. Fishkin, and M. Philipose, "Mapping and localization with RFID technology," in *Proceedings of the 2004 IEEE International Conference on Robotics and Automation (ICRA)*. New Orleans, LA, USA: IEEE, April/May 2004, pp. 1015–1020.

[2] P. Vorst and A. Zell, "Semi-autonomous learning of an RFID sensor model for mobile robot self-localization," in *European Robotics Symposium 2008*, ser. Springer Tracts in Advanced Robotics, vol. 44/2008. Springer Berlin/Heidelberg, February 2008, pp. 273–282.

[3] T. Deyle, C. C. Kemp, and M. S. Reynolds, "Probabilistic UHF RFID tag pose estimation with multiple antennas and a multipath RF propagation model," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Nice, France, Sept. 2008, pp. 1379–1384.

[4] X. Liu, M. D. Corner, and P. Shenoy, "Ferret: RFID localization for pervasive multimedia," in *Proceedings of the 8th International Conference on Ubiquitous Computing (UbiComp)*, ser. Lecture Notes in Computer Science, P. Dourish and A. Friday, Eds., vol. 4206. Irvine, CA, USA: Springer, Sept. 2006, pp. 422–440.

[5] T. Deyle, H. Nguyen, M. Reynolds, and C. C. Kemp, "RF vision: RFID receive signal strength indicator (RSSI) images for sensor fusion and mobile manipulation [sic]," in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, St. Louis, USA, Oct. 2009, pp. 5553–5560.

[6] J. Liu and M. West, *Sequential Monte Carlo Methods in Practice*, ser. Information Science and Statistics. Springer, 2001, ch. Combined Parameter and State Estimation in Simulation-based Filtering, pp. 197–224.

[7] D. Fox, "Adapting the sample size in particle filters through kld-sampling," *The International Journal of Robotics Research*, vol. 22, no. 12, pp. 985–1003, Dec. 2003.

[8] D. Joho, C. Plagemann, and W. Burgard, "Modeling RFID signal strength and tag detection for localization and mapping," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Kobe, Japan, May 2009, pp. 3160–3165.

[9] R. Liu, A. Koch, and A. Zell, "Mapping UHF RFID Tags with a Mobile Robot using 3D Sensor Model," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2013)*, Big Sight, Tokyo, Japan, November 2013.

[10] S. Thrun, "Robotic mapping: A survey," in *Exploring Artificial Intelligence in the New Millenium*, G. Lakemeyer and B. Nebel, Eds. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2002, pp. 1–35.

[11] W. Koch, "On 'negative' information in tracking and sensor data fusion: Discussion of selected examples," in *Proceedings of the Seventh International Conference on Information Fusion*, P. Svensson and J. Schubert, Eds., vol. 1. Mountain View, CA, USA: International Society of Information Fusion, June 2004, pp. 91–98.

[12] J. Hoffmann, M. Spranger, D. Göhring, M. Jüngel, and H.-D. Burkhard, "Further studies on the use of negative information in mobile robot localization," in *Proceedings of the 2006 IEEE International Conference on Robotics and Automation (ICRA)*. Orlando, Florida, USA: IEEE, May 2006, pp. 62–67.