<http://www.diva-portal.org>

Postprint

This is the accepted version of a paper presented at *IEEE International Symposium on Robot and Human Interactive Communication, RO-MAN*.

Citation for the original published paper:

Karaoguz, H., Bore, N., Folkesson, J., Jensfelt, P. (2017)
Human-Centric Partitioning of the Environment.
In: IEEE

N.B. When citing this work, cite the original published paper.

© 2017 IEEE. Personal use of this material is permitted. Permission from IEEE must be obtained for all other uses, in any current or future media, including reprinting/republishing this material for advertising or promotional purposes, creating new collective works, for resale or redistribution to servers or lists, or reuse of any copyrighted component of this work in other works.

Permanent link to this version:

<http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-215941>

Human-Centric Partitioning of the Environment

Hakan Karaoguz¹ Nils Bore¹ John Folkesson¹ and Patric Jensfelt¹

Abstract—In this paper, we present an object based approach for human-centric partitioning of the environment.

Our approach for determining the human-centric regions is to detect the objects that are commonly associated with frequent human presence. In order to detect these objects, we employ state of the art perception techniques. The detected objects are stored with their spatio-temporal information in the robot’s memory to be later used for generating the regions. The advantages of our method is that it is autonomous, requires only a small set of perceptual data and does not even require people to be present while generating the regions.

The generated regions are validated using a 1-month dataset collected in an indoor office environment. The experimental results show that although a small set of perceptual data is used, the regions are generated at densely occupied locations.

I. INTRODUCTION

Spatial partitioning of an indoor environment is important for autonomous robots to be employed in complex scenarios such as assistive-robotics, service-robotics or security-patrolling. These scenarios generally involve human-robot interactions and the understanding of human spatial concepts such as rooms, workspaces, etc.

There are several approaches for spatial partitioning of an environment. However, most of these approaches either focus on local partitioning such as single-view tabletop segmentation or room based partitioning of an entire office floor without really focusing on human presence. There are other approaches that imply human-centric partitioning of an indoor environment to detect human presence or activities [1], [2], [3]. However, these approaches mostly rely on implicitly defined manual partitions, require voluntary interactions from humans to detect the human presence or activities and too abstract to determine different activities that can take place within the same region.

In this work, we automate the process of human-centric partitioning of the environment. A human-centric partition is a region that is frequently occupied by people, usually more abstract compared to a tabletop partition and more focused than a room partition. Moreover, the human activities within these regions can be recognized with current state of the art activity recognition methods which will allow the user to associate semantic information to these regions. We refer to these semantically labeled human-centric regions as Human Activity Regions (HAR)s. As such, each HAR can represent different activities taking place within the same room.

This work is the first step to autonomously create the HARs where we employ an object based approach to capture

Fig. 1: Example of automatically generated human-centric regions (cyan) in an office room.

the densely populated regions in an office environment. Fig. 1 shows an example office room with human-centric regions determined by a mobile robot. Our way of determining these regions is to detect the objects that are related to frequent human presence and activity.

Fig. 2 shows the human presence in different indoor locations in our office environment which gives clues about the related objects. The related objects could vary from place to place and in this work it is assumed that the types of these objects are known apriori. For a factory for example, the related objects could be a workbench and the related machinery. For a hospital, it could be the patient beds and chairs. Of course, this assumption has some shortcomings as some places with frequent human presence cannot always be characterized with related objects such as a corridor. However, our results show that the generated human-centric regions cover densely populated areas in an indoor office environment.

Based on the previous work in this field, the primary contribution of our proposed method is that it autonomously performs human-centric spatial partitioning while using only a small set of perceptual data of the environment. Furthermore, we validate our approach using a dataset collected over 1-month time and against manually annotated human-centric regions based on the same definition as used in this paper. The generated regions can be used by planning algorithms to calculate optimal positions for activity recognition or by social mapping approaches to take proxemics into account [4].

¹ The authors are all with the Centre for Autonomous Systems at KTH Royal Institute of Technology, Stockholm, SE-100 44, Sweden hkarao@kth.se

Some of the important aspects of our method are; it is fully implemented as a software stack, requires minimal amount of data and it is evaluated on a real-time mobile robot platform. Thus, it can be easily deployed in scenarios with limited environmental knowledge.

The organization of this paper is as follows. The related work about spatial partitioning is introduced in Section II. In Section III and Section IV, our approach and implementation are explained in detail. The experimental results are given in Section V and the paper concludes with a brief summary in Section VI.

II. RELATED WORK

There are many studies in the literature for spatial partitioning of different environments. These studies can be grouped into two categories [5] based on the scale of the partitioned environment. The first group is the local methods where mostly single scenes are analyzed individually [6], [7], [8]. In [6], local features and geometric relationships obtained from RGB-D sensors are used to segment and semantically label scenes based on predetermined categories. Similarly in [7] an efficient way of scene segmentation is proposed using connected components and Euclidean clustering. Kunze et. al. [8] combined 3D object recognition with qualitative spatial relations between objects to segment and semantically label a tabletop scene.

The second group consists of global approaches where the aim is to partition a complete floor plan rather than single scenes [9], [10], [11], [12], [13], [14]. For the global approaches, various methods exist such as Voronoi partitioning, feature-based methods and segmentation based methods [13]. As an example, [9] uses object detection to classify the scenes as kitchen or office. Pronobis et. al. use a multi-modal probabilistic approach to semantically partition indoor environments [10]. In [11], an energy function based approach using simple spatial aspects is employed to label an office environment. Natural language descriptions along with topological and semantic representations are used to semantically label an indoor environment in [12]. In, [14] office floors are semantically partitioned into 3 categories that are office, kitchen or eating area. This approach uses the locations of recognized objects together with semantic web mining to infer the category of the scenes observed by the robot at different waypoints.

Apart from spatial partitioning approaches, there are other studies that focus on determining human presence and activities in an environment. For example, in [1], thousands of human trajectories from a 1-month dataset is used to model and predict the human presence and behaviors. For this task, the environment is manually partitioned into 12 regions and Poisson process models together with spectral analysis are employed to characterize these regions according to their activity patterns. In [3], an autonomous robot platform that is used in office environments is introduced. In order to detect the frequency of human activity, the robot waits for an input to its terminal screen from a user. Moreover, human tracks

are used to identify human activities within the manually created spatial partitions.

In [2], the human activities are recognized using Latent Semantic Analysis (LSA) together with a qualitative spatial representation. The spatial relations between the related objects and the skeleton pose estimates are used to discover semantically meaningful activity classes within manually-annotated regions. Their work uses the concept of HARs for activity recognition but all the related objects and regions are annotated manually. Our work automatize this process by first detecting the related objects and then generating the regions using the object locations and bounds.

Fig. 2: A robot's observations of different rooms.

III. METHOD

Our human-centric region generation method is composed of 3 modules, perception, memory and reasoning that are explained in Section III-A, Section III-B and Section III-C respectively. The implementation details of the method is given in Section IV.

A. Perception

This module is the key element for generating and positioning the regions precisely. When the robot performs a scan of the environment, this module detects the objects in the field of view, determines the object's location w.r.t a global frame of reference and stores the information in the memory. In practice this module will typically be case specific since the choice of objects and sensors can vary from place to place. The system can use different sensor modalities for detecting different kinds of objects.

B. Memory

The memory module stores all the perceived data over time such that it can be efficiently accessed by the reasoning module. The module is designed as a MongoDB based framework that can store the perceived information of any kind continuously in different collections with spatio-temporal information and if available meta-information such as type, instance id, etc. The stored data can be queried later

Fig. 3: Overview of the method and the implementation.

on using spatial and/or temporal constraints in an efficient manner.

C. Reasoning

The reasoning module makes use of the stored object data in the memory to generate the regions. In our method, we first determine the key and supporting objects for an environment apriori. Here, the key objects are identified as being more static such as a table, a cupboard etc. We first use the key object locations and bounds to generate the initial human-centric regions. The supporting objects on the other hand are more dynamic but generally move within a confined space such as chairs, monitors, etc. The supporting object locations and bounds are used to expand the initial key object based regions using a proximity threshold.

During region generation, we need to take into account the dynamic nature of the objects especially the supporting objects. Thus, in order to determine the most likely locations of the detected objects, clustering of object observations using the observation history is necessary. For this task, we first employ unsupervised clustering to determine the sparse object clusters in the global frame of reference. Afterwards, the centroids of the object clusters, the aggregate size of the object and an expansion parameter τ_e is used to generate the initial regions. Expansion is needed as, most of the time, human presence will not be on the object itself but around it. Thus, we want to also cover the proximity around the objects.

When the initial regions are generated based on the key objects, the supporting objects are associated with these initial regions using a proximity threshold τ_p and the final regions are generated accordingly. In case of no expansion, the initial key object based region is used as is.

IV. IMPLEMENTATION

Based on the observations of our environment with a mobile robot, tablespots and chairs are selected as the key and supporting objects respectively for human-centric region generation. Moreover, it is assumed that the robot can localize itself and object locations w.r.t a global frame of reference.

The system design of the method and the implementation is given in Fig. 3.

For the perception part, the processing steps are as follows. The robot's pan-tilt mechanism makes a sweep and collects a set of consecutive images and 3D point clouds that cover a field of approximately 360° by 180° as described in [15]. This set of data is sent to the deep object detection and tabletop detection modules for further processing. In this work, R-FCN [16] is employed to detect chairs robustly. The processing steps of object detection is given in Fig. 4 (top row). When the system receives an RGB image, R-FCN detects a chair and the corresponding depth information is used to extract the partial 3D view of the chair. As such the location of the chair w.r.t the global frame of reference can be determined. Since the robot acquires overlapping scans from each location, the same chair can be seen several times. Therefore, the detected chairs for that sweep location are compared against each other to store only the unique objects in robot's memory.

For tabletop detection, we employ a 3D depth data based approach that uses plane segmentation and convex hull generation [17]. We detect tablespots based on the knowledge that tables can be reliably characterized using a few natural rules given in Table I. We use the RANSAC based algorithm outlined in [17] to extract plane primitives from the 3D data. The planes are then filtered by checking if they fulfill the rules in Table I. The steps of this approach is shown in Fig. 4 (bottom row).

After the perception module has processed a couple of sweeps and the detected objects have been stored in the memory, we employ the reasoning module. For determining the most likely locations of the key and supporting objects, namely the tablespots and chairs, we first query the memory to retrieve the detection history. For the chair objects, the DBScan algorithm [18] is used to extract the most likely locations. The advantage of the DBScan is that it is a density based algorithm which groups neighboring detections and works with unknown number of clusters which is the case here. The output of this step is the individual object clusters

Fig. 4: The processing steps of object detection (top row) and tabletop detection (bottom row) modules.

$\bar{c}_i$, $i = 1, 2, \dots, |C|$. where $|C|$ is the total number of discovered chair clusters. For the tabletop objects, we keep track of the each detected tabletop instance t_j , $j = 1, 2, \dots, |T|$ and expand particular instance if additional information is acquired in a new sweep.

The reasoning process is outlined in Algorithm 1. The resulting human-centric regions $r_j(V_j, E_j)$ are convex polygons with vertices V_j and edges E_j .

Algorithm 1

```

1:  $\mathcal{T}$ : The set of detected table tops
2:  $\mathcal{C}$ : The set of detected chair clusters
3:  $\mathcal{R}$ : The set of generated human-centric regions
4:  $\mathcal{R} \leftarrow \emptyset$ 
5: for all  $t_j$  such that  $t_j \in \mathcal{T}$  do
6:    $r_{t_j} \leftarrow \text{generateinitialregion}(t_j, \tau_e)$ 
7: end for
8: for all  $\bar{c}_i$  such that  $\bar{c}_i \in \mathcal{C}$  do
9:    $r_{c_i} \leftarrow \text{generateinitialregion}(\bar{c}_i, \tau_e)$ 
10: end for
11: for all  $r_{t_j}$  do
12:   for all  $r_{c_i}$  do
13:     if  $\text{associatesupportingobject}(r_{t_j}, r_{c_i})$  then
14:        $r_{t_j} \leftarrow \text{expandregion}(r_{t_j}, r_{c_i}, \tau_p)$ 
15:     end if
16:   end for
17:    $\mathcal{R} \leftarrow \mathcal{R} \cup r_{t_j}$ 
18: end for
19: return( $\mathcal{R}$ )

```

V. EXPERIMENTS

A. Setup

Experiments are conducted using the dataset collected at the Centre for Autonomous Systems at KTH using a Scitos G5 robot with an RGB-D camera on a pan-tilt unit [19].

The data contains RGB-D scans over a period of 1-month. During its tour, the robot patrols an office floor including kitchen, meeting room and two office rooms. The robot used gmapping [20] for 2D mapping of the environment and the AMCL [21] algorithm for localization. In total 720 sweeps are collected over the whole time period. Table I shows the used parameters for these experiments.

TABLE I: Parameters used for human-centric region generation.

Parameter	Value
Tabletop Angle	17°
Tabletop Height	0.6-1.1 meters
Tabletop Min Area	0.3 m ²
τ_d	0.5 meters
τ_p	1 meters
τ_e	0.1

B. Results

To serve as a reference in the evaluation of the automatically generated regions, the environment is first manually partitioned by using the true tabletop and chair locations. The manual inspection of the environment resulted in 14 regions which are shown in Fig. 5 (left column).

1) *Human-Centric Region Generation*: For the generation of the human-centric regions, we have only used 16 sweeps which corresponds to 2% of the dataset. On average, each room was visited 4 different times of a day. The reason for visiting rooms at different times is to capture the variability of the chair locations and detect and refine the static table tops with varying clutter.

If we look at the performance of the perception module, 34 chair objects were detected in total. The DBScan¹ algorithm determined 11 chair clusters. The analysis of these cluster

¹DBScan was initialized with parameters minimum sample size of 2 and ϵ 0.6 meters

Fig. 5: Regions generated in each different room. The left column shows the manually annotated regions based on true tabletop and chair locations. The middle column shows the initial regions that are generated using chairs (red) and tabletops (green). Right column shows the final regions that are generated after the association of chairs and tables.

locations shows that all of them correspond to an actual chair location. In total 11 out of 17 actual chairs were identified. The low true-positive rate for chair detection is probably due to the algorithm having problems recognizing occupied or occluded chairs. Nevertheless, the undetected chairs do not affect the number of generated regions.

The tabletop detection module also worked robustly when the tables are large enough, even if cluttered. This module has detected and tracked 16 tables. 15 of them were actually correspond to a tabletop while 1 of them was the top part of a safebox in the meeting room. The advantage of detecting tabletops compared to chairs is that they are mostly static, simpler in shape and larger in size.

Fig. 5 shows the manually and automatically generated regions in each room. The middle column shows the initial

regions where the red are chair based while the green are tabletop based. If we look at the initial tabletop regions, it is seen that all the workplaces, meeting room table, kitchen counter and tables were successfully determined. However, there was one extra tabletop region in the meeting room where a safebox was perceived as a tabletop because of its height and flat top surface. Another extra tabletop was detected in *Office2* which was not annotated during manual inspection. This is due to the fact that the particular table is only used for storage purposes and is not usually occupied. However, this information is unknown to the robot so a region is automatically generated around this table. When the initial chair and tabletop regions were associated, 7 of the initial regions were expanded. In total 16 regions were generated automatically.

The results show that auto-generated regions are located in all the places marked by the human annotator. The exceptions are the safebox in the meeting room and the tabletop in *Office2*, the removal of which requires high level knowledge not possessed by the robot.

It is also observed that the manually annotated regions are somewhat coarser than the auto-generated regions with our choice of τ_e . This is due to the fact that the human annotator can determine all the true locations of tables and chairs from the data while the auto-generated regions rely on robot's perceptual capability. Thus, the auto-generated regions are formed only around perceived tables and chairs.

Fig. 6: The effect of long-term environmental dynamics in Office1. Left: regions generated with limited data. Right: regions generated with long-term data.

2) The Effects of Long-Term Environmental Dynamics:

In order to analyze the effects of long-term environmental dynamics such as movement/disappearance of objects, etc. we applied our approach to the whole dataset with the same parameters. As a result, the approach found 13 chair clusters and 16 tablespots which were location-wise very similar to the previous results we got with the limited data. 9 of these initial tabletop regions were expanded after association with chairs.

The analysis of the final regions showed that, most of them were mostly similar to the ones that were generated with limited data. However, it is observed that the tabletop areas were captured more accurately probably due to the increased number of observations which can be seen in Fig. 6 for *Office1*. Moreover, a region in *Office1* is expanded drastically due to the large chair cluster formed in between the two tables. These kinds of variations can occur in long-term scenarios since the people can move between tables with their chairs in small rooms. The results of this analysis show that our approach can accurately create human-centric regions even with a small set of data and these regions are robust to environmental dynamics over time.

3) *People Presence in Human-Centric Regions:* As a third evaluation we analyze whether the automatically generated human-centric regions are occupied frequently by people or not. For this task, we again employed the deep-learning based object detection module explained in Section IV which can detect and localize people from RGB-D data. The person detections were done using the same dataset that is composed of the sweeps of the environment at regular intervals each

Fig. 7: Auto-generated regions superimposed on people density map. The red areas represent the high density locations while blue areas represent low density locations. Purple colored dots represent the people detections while green spheres represent the local density peaks.

day. In total 1161 human detections were made in the whole office floor.

We then performed the people density estimation of the office floor by first modeling the people presence at each detection location as a Gaussian density function with $\sigma^2 = 1.0 \text{ m}^2$. Secondly, we accumulated the densities for each human detection and created a people density map that is shown in Fig. 7. Here the red areas are the locations with higher density while the blue areas have lower density. Gray areas are unobserved. Purple colored dots represent the people observations.

The density map shows that the most populated regions are the offices and the kitchen. The meeting room has less density because of the lack of observations in that room. In the corridor, there are plenty of people detections but the overall density is low since the people detections are sparse in this region.

In order to measure whether the dense areas are actually covered by the human-centric regions, we first identified the local peaks from the density map with the constraint that peaks have to be at least 2 meters apart. As such we can locate as many sparse peaks as possible. As a result, 9 peaks which are shown as green spheres in Fig. 7 are found. It is observed that 6 of these peaks were located within the auto-generated regions while 2 of them were in close proximity of a region. The result shows that nearly 90% of the dense

locations are found within or in close proximity of the auto-generated human-centric regions.

Moreover, this result verifies the idea of using objects associated with frequent human presence to identify densely populated human-centric regions. These densely populated human-centric regions can later be transformed into HARs by employing activity recognition methods and can replace the manually annotated regions for the human activity recognition approach given in [2].

VI. CONCLUSION AND FUTURE WORK

In this paper we introduced the concept of human-centric partitioning of the environment and proposed an object based approach for automatically generating these partitions. It is shown that our approach can accurately generate human-centric regions with a very small set of perceptual data.

Furthermore, we analyzed the effects of long-term environmental dynamics with a dataset collected over a 1-month time. It is observed that the dynamics do not dramatically affect the regions created with a small set of perceptual data.

The auto-generated regions are also validated in terms of people density coverage. The results show that the regions cover most of the populated areas which also verify the idea of using objects that are related to frequent human presence for region generation.

As a future work, we plan to combine our method with human activity recognition approaches to actually form the HARs. Thus, the system will be able to associate semantic information to these human-centric regions. We will also work on the perception module to improve the detection of different objects that could be used for generating human-centric regions in different environments such as a factory or a shop.

VII. ACKNOWLEDGEMENTS

The work in this paper is funded by the EU project STRANDS (ICT-2011-600623), the SSF project FACT and the Centre for Autonomous Systems.

REFERENCES

- [1] F. Jovan, J. Wyatt, N. Hawes, and T. Krajník, "A poisson-spectral model for modelling the temporal patterns in human data observed by a robot," in *IEEE/RSJ International Conference on Intelligent Robots and Systems 2016*, October 2016.
- [2] P. Duckworth, M. Alomari, Y. Gatsoulis, D. Hogg, and A. Cohn, "Unsupervised activity recognition using latent semantic analysis on a mobile robot," Amsterdam, Netherlands, pp. 1062–1070, August 2016.
- [3] N. Hawes, C. Burbridge, F. Jovan, L. Kunze, B. Lacerda, L. Mudrová, J. Young, J. L. Wyatt, D. Hebesberger, T. Körtner, R. Ambrus, N. Bore, J. Folkesson, P. Jensfelt, L. Beyer, A. Hermans, B. Leibe, A. Aldoma, T. Faulhammer, M. Zillich, M. Vincze, M. Al-Omari, E. Chinellato, P. Duckworth, Y. Gatsoulis, D. C. Hogg, A. G. Cohn, C. Dondrup, J. P. Fentanes, T. Krajník, J. M. Santos, T. Duckett, and M. Hanheide, "The STRANDS project: Long-term autonomy in everyday environments," *CoRR*, vol. abs/1604.04384, 2016.
- [4] K. Charalampous, I. Kostavelis, and A. Gasteratos, "Robot navigation in large-scale social maps: An action recognition approach," *Expert Systems with Applications*, vol. 66, pp. 261–273, 2016.
- [5] I. Kostavelis and A. Gasteratos, "Semantic mapping for mobile robotics tasks: A survey," *Robotics and Autonomous Systems*, vol. 66, pp. 86–103, 2015.
- [6] H. S. Koppula, A. Anand, T. Joachims, and A. Saxena, "Semantic Labeling of 3D Point Clouds for Indoor Scenes," *Neural Information Processing Systems*, pp. 1–9, 2011.
- [7] A. J. Trevor, S. Gedikli, R. B. Rusu, and H. I. Christensen, "Efficient organized point cloud segmentation with connected components," *Semantic Perception Mapping and Exploration (SPME)*, 2013.
- [8] L. Kunze, C. Burbridge, M. Alberti, A. Tippur, J. Folkesson, P. Jensfelt, and N. Hawes, "Combining top-down spatial reasoning and bottom-up object class recognition for scene understanding," in *IEEE International Conference on Intelligent Robots and Systems*, 2014, pp. 2910–2915.
- [9] P. Viswanathan, T. Southey, J. J. Little, and A. Mackworth, "Automated place classification using object detection," in *Computer and Robot Vision (CRV), 2010 Canadian Conference on*, 2010, pp. 324–330.
- [10] L. Pronobis and P. Jensfelt, "Large-scale semantic mapping and reasoning with heterogeneous modalities," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 3515–3522.
- [11] K. Sjöö, "Semantic map segmentation using function-based energy maximization," in *Proceedings - IEEE International Conference on Robotics and Automation*, 2012, pp. 4066–4073.
- [12] S. Hemachandra, M. R. Walter, S. Tellex, and S. Teller, "Learning Spatially-Semantic Representations from Natural Language Descriptions and Scene Classification," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2014, pp. 2623–2630.
- [13] R. Bormann, F. Jordan, W. Li, J. Hampp, and M. Haegle, "Room Segmentation: Survey, Implementation, and Analysis," in *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, 2016, pp. 1019–1026.
- [14] J. Young, V. Basile, M. Suchi, L. Kunze, N. Hawes, M. Vincze, and B. Caputo, "Making sense of indoor spaces using semantic web mining and situated robot perception," in *1st International Workshop on Application of Semantic Web technologies in Robotics (AnSWeR)*, 2017.
- [15] R. Ambrus, N. Bore, J. Folkesson, and P. Jensfelt, "Meta-rooms: Building and maintaining long term spatial models in a dynamic world," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept 2014, pp. 1854–1861.
- [16] D. Jifeng, L. Yi, H. Kaiming, and J. Sun, "R-FCN: Object detection via region-based fully convolutional networks," *arXiv preprint arXiv:1605.06409*, 2016.
- [17] R. Schnabel, R. Wahl, and R. Klein, "Efficient ransac for point-cloud shape detection," *Computer Graphics Forum*, vol. 26, no. 2, pp. 214–226, Jun. 2007.
- [18] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise."
- [19] R. Ambrus, J. Ekekrantz, J. Folkesson, and P. Jensfelt, "Unsupervised learning of spatial-temporal models of objects in a long-term autonomy scenario," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE, 2015, pp. 5678–5685.
- [20] G. Grisetti, C. Stachniss, and W. Burgard, "Improved techniques for grid mapping with rao-blackwellized particle filters," *Trans. Rob.*, vol. 23, no. 1, pp. 34–46, Feb. 2007.
- [21] 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, 2003.