Active Particle Filter Networks: Efficient Active Localization in Continuous Action Spaces and Large Maps

Daniel Honerkamp, Suresh Guttikonda, and Abhinav Valada

Abstract— **Accurate localization is a critical requirement for most robotic tasks. The main body of existing work is focused on passive localization in which the motions of the robot are assumed given, abstracting from their influence on sampling informative observations. While recent work has shown the benefits of learning motions to disambiguate the robot's poses, these methods are restricted to granular discrete actions and directly depend on the size of the global map. We propose Active Particle Filter Networks (APFN), an approach that only relies on local information for both the likelihood evaluation as well as the decision making. To do so, we couple differentiable particle filters with a reinforcement learning agent that attends to the most relevant parts of the map. The resulting approach inherits the computational benefits of particle filters and can directly act in continuous action spaces while remaining fully differentiable and thereby end-to-end optimizable as well as agnostic to the input modality. We demonstrate the benefits of our approach with extensive experiments in photorealistic indoor environments built from real-world 3D scanned apartments. Videos and code are available at `http://apfn.cs.uni-freiburg.de`.**

I. Introduction

The ability of a robot to accurately localize itself is a core requirement across almost all robotic tasks from navigation [1], [2] to mobile manipulation [3], [4], [5]. Accordingly, a broad body of research has been devoted to this topic. The by far most common approach is to first define an initial guess of the robot's pose, then manually move the robot until the localization algorithm has roughly converged and continue to constantly localize the robot while it executes its tasks. This is known as passive, local localization.

Most localization algorithms rely on a form of feature matching between the current observations and a given (2D) map of the environment. As such their performance strongly depends on the current observations, which in turn are decided by the robot's motions which decide what parts of the map will be observed. But the ability to sample informative observations has remained largely unexplored. In this work, we investigate the benefits of *active localization*, in which the robot can actively seek observations that are most informative of its current pose in the environment. Furthermore, the agent can counteract the strengths and weaknesses of particular localization modules by actively avoiding ambiguous situations and failure modes of the localization module.

Previous work has extended Adaptive Markov Localization to active control by greedily maximizing information theoretic quantities [6], [7], but for the most part, remained restricted to analytical observation models and structured observations. More recently, learning-based methods have shown the benefits of active decision making for localization [8], [9], though have remained constrained to simple environments [8] or discrete actions and small maps [9], having to process the global map at every possible orientation of the agent at each step.

We couple probabilistic and learning-based methods through learned particle filters [10] and deep reinforcement learning (RL) to generalize to continuous action spaces and arbitrary sensor modalities independent of map size. Particle filters [11] enable efficient representation of multi-modal beliefs over large maps. These mechanisms can be made fully differentiable [10], [12], enabling us to learn the components of a particle filter end-to-end, thereby extending it to abstract observations such as pixels or depth maps. Importantly, these networks only need to process local information for each particle. We then train a reinforcement learning agent that selects actions to minimize the overall localization error, following the same principle of processing only local information over the most likely hypotheses through a hard attention mechanism. In contrast to previous work, this enables us to process hypotheses over continuous poses [8], [9] while at the same time breaking the dependency on processing the full map with a neural network.

We evaluate our approach in extensive photorealistic scenes of real-world 3D scanned apartments from the gibson dataset [13] in the iGibson simulator [14] and find substantial improvements in localization error over the baselines, demonstrating the benefits of the learned policy.

II. Related Work

Passive Localization: Established localization heavily rely on Bayesian filtering-based techniques such as Kalman filters [15] which are restricted to modeling unimodal (Gaussian) beliefs, Multi-Hypothesis Kalman filters that use mixtures of Gaussians [16] or non-parametric particle filters which can model arbitrary distributions. Particle filters are widely used in methods such as Monte Carlo Localization and Adaptive Monte Carlo Localization (AMCL) [11]. Though these methods usually rely on structured observations and analytic observation models and therefore are most commonly used with LiDAR observations. While there are approaches that incorporate depth or camera images [17], [18], constructing observation models for them is challenging. Recently, fully differentiable versions of particle filters have been introduced [10], [12]. These fully differentiable

Department of Computer Science, University of Freiburg, Germany. This work was funded by the European Union's Horizon 2020 research and innovation program under grant agreement No 871449-OpenDR.

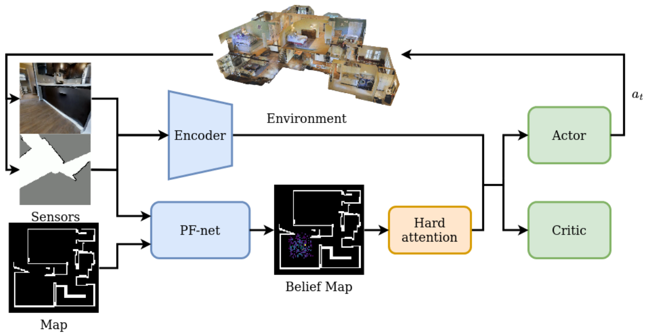

Fig. 1: Illustration of our proposed Active Particle Filter Networks.

versions enable the use of arbitrary modalities through end-to-end optimization. Learning-based methods have also been proposed to learn circular features [19], extract explicit features such as room layout edges [20] or to estimate odometry directly from visual inputs [21], [22].

Active Localization: Active localization has received comparably little attention in the past. Active versions of both Markov Localization [6], [7] and Kalman filters [23] have been proposed. These methods inherit the need for structured observations or expert-specified observation models and aim to maximize information theoretic quantities such as the reduction in entropy of the belief. Chaplot *et al.* [8] introduce a learnable Bayesian filtering approach in combination with reinforcement learning. The model relies on access to observations from across the environment to compute features ahead of time and at each step has to process the full map for every possible discrete orientation. As a consequence, the approach does not easily generalize to different map sizes at test time and does not scale well to large maps or continuous actions. Gottipati *et al.* [9] introduce a hierarchical likelihood model in which the full map is processed at a coarse resolution and only likely areas are processed at higher resolutions. But the dependency on the map size remains and only discrete actions can be evaluated. For both approaches, the dimensionality of the reinforcement learning agent's inputs scales linearly with the discretization of the rotation actions. In contrast, our approach never has to process the full map with a neural network and can directly evaluate continuous poses and actions.

III. Active Particle Filter Networks

A. Problem Statement

We assume a mobile robot that receives exteroceptive sensor readings $sens_t$ and proprioceptive odometry measurements m_t, placed randomly in an environment. Given a map M of the environment, we seek the sequence of actions $a_{1:T}$ that minimizes the pose error of the robot over a fixed time horizon T. We may be given an initial guess of the initial pose of the robot (local localization) or have to start from a uniform belief over the full map (global localization). An overview of our approach is depicted in Figure 1.

B. Localization Module

The robot starts with an initial belief b_0, either uniformly distributed over the map or based on an initial guess. Given the current observation $o_t = [sens_t, m_t]$, we then use a differentiable particle filter network (PF-net) [10] to update the current belief over the robot's pose. PF-net uses neural networks to present the observation and transition model of a particle filter. By using a soft-resampling, where new particle weights $w_t^{\prime k}$ of K particles are sampled from a distribution

$$q(k) = \alpha w_t^k + (1-\alpha)/K \tag{1}$$

the gradients are non-zero for values of $\alpha \neq 1$, enabling us to optimize through the whole network. The observation model calculates the likelihood f_t^k of a particle based on an encoding of the current sensor readings and particle-centric local map which is extracted from the global map through a differentiable spatial transformer module [24]. As a result, the likelihood of each particle can be evaluated based on local information without the need to process the full global map. This provides a number of advantages for active localization: (i) the network is fully differentiable and thereby can be jointly optimized with deep reinforcement learning algorithms, (ii) it is flexible to arbitrary robot sensors, making it applicable to a wide range of robotic platforms and (iii) it can handle continuous actions and arbitrary map sizes. The model is trained end-to-end to minimize the mean squared pose error. We follow the architecture of the original work [10].

C. Active Localization

We aim to learn a policy to move the agent such that, given the current belief about the robot's pose and the localization module, it can best disambiguate the true pose. The agent is operating in a Partially Observable Markov Decision Process (POMDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{O}, \mathcal{T}(s'|s,a), P(o|s), r(s,a))$ where $\mathcal{S}, \mathcal{O}$ and $\mathcal{A}$ are the state, observation and action spaces, $\mathcal{T}$ and P describe the transition and observation probabilities, and r and γ are the reward and discount factor. The agent's objective is to learn a policy $\pi(a|\cdot)$ that maximises the expected return $\mathbb{E}_\pi[\sum_{t=1}^{T} \gamma^t r(s_t, a_t)]$.

Belief Representation: The PF-net provides us with a multi-modal belief b_t over the global map, represented by the particle state. We transform this into a spatial, permutation invariant representation by projecting the particles into a belief map of dimension $H \times W \times 4$ where the first channel is the occupancy map, the second channel is the aggregated weights for all particles in a given cell and the third and fourth channel are the weighted sine and cosine of all particles in a given cell. The sine and cosine are used to circumvent the non-linearity in the angles.

Agent: We propose a reinforcement learning agent that observes both its current belief together with the low-level robot observations and learns a policy $\pi(a_t|b_t, o_t; l)$ where l is the localization module. This allows it to improve the localization in two ways: (i) actively sample the most informative

Parameter	PF-net	RL
train steps	400,000	1,000,000
batch size	8	256
lr	$2.5\mathrm{e}{-3}$	$3\mathrm{e}{-4}$
resample	false	true
β	0.36	0.36
T	25	50
particles	30	500
initial distribution	tracking	semi-global
initial std (translation, angular)	0.3, $\pi/6$	0.3, $\pi/6$
transition noise (translation, angular)	0.0, 0.0	0.01, $\pi/36$
RL only		
control frequency, replay buffer size	1.7Hz	50,000
τ, γ	0.005	0.99
α, $\lambda_{collision}$	0.5	0.1

TABLE I: Training hyperparameters for the PF-net and RL agent.

sensor readings and (ii) take into account the localization module's strengths and weaknesses, e.g. avoid observations where the localization module does not perform well.

While the belief stretches the full map, within very few steps the particles concentrate on a small number of most likely regions. We apply the principle of local information to break the dependency on the full map. We extract local maps around the modes of the particle distribution from the belief map. The agent then observes a stack of k local belief maps, each centered and oriented according to a mode of the distribution. This is akin to a hard attention mechanism, which can be made fully differentiable if desired [25]. In practice, just using the mean position and orientation of the particles works well, but extending this to cover the top k modes is straightforward. In contrast to previous work, this allows to process and generalize to arbitrary map sizes and arbitrary continuous poses and actions.

Training: The agent is trained to directly minimize the prediction error of the localization network. At each step, it receives a reward $r = -\mathcal{L}_{pfnet} - \lambda_{collision} * \mathbb{1}_{collision}$ where $\mathcal{L}_{pfnet}$ is the prediction loss of the PF-net, $\mathbb{1}_{collision}$ is a binary collision indicator and $\lambda_{collision}$ is a weighting constant. The agent has a fixed number of environment steps to localize itself, after which the episode terminates. While the approach is fully differentiable and can be optimized end-to-end, we find it beneficial to pretrain the localization network for better stability. Though joint finetuning may be able to further improve results. For pretraining we use a goal-reaching agent (see Section IV-A) to collect a dataset of 4,000 episodes of length 25 and then perform supervised training following Karkus *et al.* [10], using a tracking task with only 30 particles. We train the RL agent with soft-actor critic (SAC) [26], which has been shown to produce strong policies in continuous control and robotics tasks. Hyperparameters are reported in Table I.

IV. Experimental Results

A. Experiment Setup

To evaluate our approach, we train a LoCoBot robot in the photorealistic iGibson simulator [14]. The LoCoBot robot has a differential drive and is equipped with an RGB-D

Fig. 2: Tracking (top), semi-global (mid) and global (bottom) localization tasks. From left: a) initial particle distribution b) global map and trajectory with estimated (green) and ground truth (red) poses at each step. Circles denote the final poses. c) local belief map observed by the RL agent d) occupancy grid e) RGB and f) depth.

camera with a field-of-view of 90° and a maximum depth of $10\,\mathrm{m}$ as well as a LiDAR with a range of 240°. The action space consists of the linear and angular velocities for the base. We use a subset of 45 apartment scenes from the gibson dataset [13], split into 38 training and 7 unseen test apartments. The test apartments are completely unseen by both the PF-net and the RL agent.

Baselines: We compare against the following baselines:

- *Avoid:* A simple heuristic policy that drives forward until its depth camera recognizes a close object, then drives backwards or turns away from the obstacle.
- *Goalnav:* A policy that navigates towards a random target using a path-planner with access to the ground truth traversability map and robot pose.
- *Turn:* An agent that always turns in place at maximum angular velocity.

Tasks: We focus on three localization tasks, ranging from local to global localization. These are

- *Tracking:* the initial particles are sampled from a multi-variate Gaussian distribution with a standard deviation of $0.3\,\mathrm{m}$ and 30° and centered at a random pose sampled with the same standard deviations around the ground truth robot pose. The PF-net uses 300 particles.
- *Semi-global localization:* We uniformly sample 500 particles in a box of $3.3{\times}3.3\,\mathrm{m}$ around the initial guess.
- *Global localization:* We sample 3,000 particles uniformly across the traversable area of the whole map.

Metrics: We report the root mean squared positional error in centimeters and root mean squared angular error in radians, referred to as *position* and *orient* in the tables. All metrics are averaged over 50 episodes.

B. Passive Localization

The original PF-net model has focused on evaluation in the simpler House3D dataset [27]. We implement a version of this model for scenes from the photorealistic gibson dataset, which are based on real-world 3D scans of apartments. We report the results for passive localization for different modalities based on the goalnav agent that collected the

Task	seen						unseen					
	Tracking		Semi-Global		Global		Tracking		Semi-Global		Global	
Modality	position	orient	position	orient	position	orient	position	orient	position	orient	position	orient
LiDAR	20.8	0.13	27.2	0.21	111.4	0.33	18.9	0.12	23.7	0.16	141.2	0.38
RGB-D	24.8	0.15	30.2	0.20	126.6	0.30	24.5	0.16	29.2	0.18	144.1	0.34

TABLE II: Passive localization results on the iGibson dataset for different localization tasks. We report the average root mean squared positional error in centimeter (*position*) and the root mean squared orientation error of the robot's yaw in radians (*orient*). Evaluated with the pretraining settings for $T = 25$.

Task	seen						unseen					
	Tracking		Semi-Global		Global		Tracking		Semi-Global		Global	
Agent	position	orient	position	orient	position	orient	position	orient	position	orient	position	orient
Goalnav	16.8	0.12	18.2	0.12	99.3	0.24	14.9	0.11	21.4	0.15	113.3	0.21
Avoid	15.8	0.13	22.4	0.15	152.0	0.32	15.8	0.12	33.5	0.19	162.9	0.29
Turn	**11.8**	0.80	14.6	0.09	103.1	0.30	13.9	0.10	19.8	0.12	115.9	0.31
APFN (ours)	13.4	**0.10**	**11.7**	**0.08**	**74.8**	**0.16**	**11.1**	**0.08**	**16.3**	**0.11**	**63.3**	**0.17**

TABLE III: Active localization results in the iGibson simulator in seen and unseen apartments. The PF-net uses the LiDAR modality. We report the average root mean squared positional error in centimeter (*position*) and the root mean squared orientation error of the robot's yaw in radians (*orient*).

training data in Table II. LiDAR scans are converted to occupancy maps in which 0 is free space, 1 unexplored, and 2 occupied. The PF-net performs well in these more complex scenes, achieving a positional error of around $20\text{-}25\,\mathrm{cm}$ for tracking, which, for both modalities, is actually lower than the $40 - 49\,\mathrm{cm}$ error reported on the House3D dataset [10]. Moreover, the network generalizes well to unseen apartments, showing no significant generalization gap. Even though the field-of-view of the RGB-D camera is much smaller than what the LiDAR can sense, both modalities achieve relatively similar performance, highlighting that the network is able to extract rich information in the complex pixel observations.

C. Active Localization

We focus on the best performing LiDAR modality. The RL agent observes the robot state consisting of current forward and angular velocities, a collision flag, and the remaining steps in the episode together with the LiDAR and the local belief map. For obstacle avoidance, it also receives the RGB-D observations. The policy consists of a shared feature encoder, made up of three convolutional networks for RGB, depth and LiDAR. Each network consists of layers with (channels, kernel size, stride) of [(32, (3, 3), 2), (64, (3, 3), 2), (64, (3, 3), 1), (64, (2, 2), 1)]. These features are then concatenated with the robot state and passed to the actor and critic, consisting of a two-layer MLP with 512 neurons. All intermediate layers are followed by ReLU activations. All pixel-based observations are of size 56×56. While we train the PF-net on ground truth odometry data, during the policy training we add zero-centered Gaussian noise with standard deviation of $1\,\mathrm{cm}$ and 5° to the transitions. We train the policy with 500 particles and at test time evaluate it with varying numbers of particles according to the task definitions.

Table III reports the results for the active localization tasks for both seen and unseen apartments. We find large differences in localization performance across the different motion models. This highlights the strong dependence on the robot's movements and confirms the importance of active decision making. APFN consistently achieves the best localization across all tasks. The only exception is the positional error in the tracking task, in which the turn policy achieves a very low positional error, but suffers from a large angular error. Moreover, the agent successfully generalizes to unseen apartments. Note that these apartments have not been seen by both the RL agent and the localization module. To succeed in these apartments, the agent has to learn general movement patterns and the ability to seek out informative regions. Lastly, we find that differences in localization performance are particularly large in the global localization task with our approach reducing the positional error by over 60% in comparison to other motions. This is expected as in global localization we have the least amount of prior information about the robot's pose. Qualitatively, the agent performs targeted movements with frequent rotations that reveal a large area of the apartments. Examples are shown in Figure 2.

V. Conclusion

We introduce APFN which combine probabilistic filtering methods with learned decision making to accurately localize a robot in realistic indoor environments. In contrast to previous methods, our approach scales to continuous action spaces and arbitrary map sizes by selectively attending to only local information. We evaluate this ability in photorealistic indoor environments and find that it is able to accurately localize itself in both seen and completely unseen apartments and considerably outperforms the baselines. In future work, we aim to incorporate simultaneous control of sensors such as actuated cameras. Another promising avenue is the extension of learning-based localization and attention mechanisms to dynamic environments and noisy, partial or incorrect maps in which it becomes important to selectively filter out uncertain or incorrect observations.

References

[1] J. V. Hurtado, L. Londoño, and A. Valada, "From learning to relearning: A framework for diminishing bias in social robot navigation," *Frontiers in Robotics and AI*, vol. 8, p. 650325, 2021.

[2] A. Younes, D. Honerkamp, T. Welschehold, and A. Valada, "Catch me if you hear me: Audio-visual navigation in complex unmapped environments with moving sounds," *arXiv preprint arXiv:2111.14843*, 2021.

[3] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237–267, 2002.

[4] D. Honerkamp, T. Welschehold, and A. Valada, "$\mathrm{N}^2\mathrm{m}^2$: Learning navigation for arbitrary mobile manipulation motions in unseen and dynamic environments," *arXiv preprint arXiv:2206.08737*, 2022.

[5] ——, "Learning kinematic feasibility for mobile manipulation through deep reinforcement learning," *IEEE Robotics and Automation Letters*, 2021.

[6] D. Fox, W. Burgard, and S. Thrun, "Active markov localization for mobile robots," *Robotics and Autonomous Systems*, vol. 25, no. 3-4, pp. 195–207, 1998.

[7] ——, "Markov localization for mobile robots in dynamic environments," *Journal of Artificial Intelligence Research*, vol. 11, pp. 391–427, 1999.

[8] D. S. Chaplot, E. Parisotto, and R. Salakhutdinov, "Active neural localization," *arXiv preprint arXiv:1801.08214*, 2018.

[9] S. K. Gottipati, K. Seo, D. Bhatt, V. Mai, K. Murthy, and L. Paull, "Deep active localization," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4394–4401, 2019.

[10] P. Karkus, D. Hsu, and W. S. Lee, "Particle filter networks with application to visual localization," in *Proc. of the Conference on Robot Learning*, 2018, pp. 169–178.

[11] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte carlo localization for mobile robots," *Artificial intelligence*, vol. 128, no. 1-2, pp. 99–141, 2001.

[12] R. Jonschkowski, D. Rastogi, and O. Brock, "Differentiable particle filters: End-to-end learning with algorithmic priors," in *Robotics: Science and Systems*, Pittsburgh, Pennsylvania, June 2018.

[13] F. Xia, A. R. Zamir, Z. He, A. Sax, J. Malik, and S. Savarese, "Gibson env: Real-world perception for embodied agents," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2018, pp. 9068–9079.

[14] C. Li, F. Xia, R. Martín-Martín, M. Lingelbach, S. Srivastava, B. Shen, K. E. Vainio, C. Gokmen, G. Dharan, T. Jain, A. Kurenkov, C. K. Liu, H. Gweon, J. Wu, L. Fei-Fei, and S. Savarese, "igibson 2.0: Object-centric simulation for robot learning of everyday household tasks," in *2021 Conference on Robot Learning (CoRL)*, ser. Proceedings of Machine Learning Research, vol. 164, 2021, pp. 455–465.

[15] R. Smith, M. Self, and P. Cheeseman, "Estimating uncertain spatial relationships in robotics," in *Autonomous robot vehicles*, 1990, pp. 167–193.

[16] I. J. Cox and J. J. Leonard, "Modeling a dynamic environment using a bayesian multiple hypothesis approach," *Artificial Intelligence*, vol. 66, no. 2, pp. 311–344, 1994.

[17] F. Dellaert, W. Burgard, D. Fox, and S. Thrun, "Using the condensation algorithm for robust, vision-based mobile robot localization," in *Proceedings of the IEEE computer society conference on computer vision and pattern recognition*, vol. 2, 1999, pp. 588–594.

[18] B. Coltin and M. Veloso, "Multi-observation sensor resetting localization with ambiguous landmarks," *Autonomous Robots*, vol. 35, no. 2, pp. 221–237, 2013.

[19] Z. Min, N. Khosravan, Z. Bessinger, M. Narayana, S. B. Kang, E. Dunn, and I. Boyadzhiev, "Laser: Latent space rendering for 2d visual localization," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 2022, pp. 11 122–11 131.

[20] F. Boniardi, A. Valada, R. Mohan, T. Caselitz, and W. Burgard, "Robot localization in floor plans using a room layout edge extraction network," in *Int. Conf. on Intelligent Robots and Systems*, 2019, pp. 5291–5297.

[21] R. Clark, S. Wang, H. Wen, A. Markham, and N. Trigoni, "Vinet: Visual-inertial odometry as a sequence-to-sequence learning problem," in *Proc. of the National Conference on Artificial Intelligence*, vol. 31, no. 1, 2017.

[22] N. Vödisch, D. Cattaneo, W. Burgard, and A. Valada, "Continual slam: Beyond lifelong simultaneous localization and mapping through continual learning," *arXiv preprint arXiv:2203.01578*, 2022.

[23] P. Jensfelt and S. Kristensen, "Active global localization for a mobile robot using multiple hypothesis tracking," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 5, pp. 748–760, 2001.

[24] M. Jaderberg, K. Simonyan, A. Zisserman *et al.*, "Spatial transformer networks," *Advances in neural information processing systems*, vol. 28, 2015.

[25] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *Int. Conf. on Machine Learning*, 2015, pp. 2048–2057.

[26] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Int. Conf. on Machine Learning*, vol. 80, 2018, pp. 1861–1870.

[27] Y. Wu, Y. Wu, G. Gkioxari, and Y. Tian, "Building generalizable agents with a realistic and rich 3d environment," *arXiv preprint arXiv:1801.02209*, 2018.