2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC)

DQN-based Coverage Maximization for Mobile Video Camera Networks

Seungho Lee, Si Young Jang, Soon J. Hyun, and Dongman Lee*
KAIST, School of Computing
Daejeon, Republic of Korea
{konglsh, sy.jang, sjhyun, dlee}@kaist.ac.kr

2021 IEEE 18th Annual Consumer Communications & Networking Conference (CCNC) | 978-1-7281-9794-4/21/$31.00 ©2021 IEEE | DOI: 10.1109/CCNC49032.2021.9369618

***Abstract*—Coverage maximization is an important issue of Mobile wireless sensor networks (M-WSN). Especially for visual sensors like video camera which have a specific sensing direction and range, obstacles and the position of the sensors should also be considered. In this paper, we propose an efficient coverage maximization method for mobile video camera networks by leveraging DQN, a deep reinforcement learning algorithm. Evaluation results show that the proposed method can cover up to 4.93% better than existing ones.**

I. Introduction

The advent of Internet of Things (IoT) technology enables many applications to continuously sense environment states like temperature, brightness, sound or visual information and process them more intelligently. Moreover, it is no longer unrealistic to witness a network of mobile wireless IoT sensors like drones, surveillance robots, autonomous cars, etc play a major role in this. One key challenge in managing a visual sensor based mobile wireless sensor network (M-WSN) is how to make a given number of sensors cover as large area as possible (coverage maximization), in the presence of obstacles.

Unlike most other types of sensors, a camera has a direction of observation, that is, its sensing coverage is not circular but fan-shaped. The sensing range of a camera called a field of view (FoV), which has a fan or triangular shape, is determined according to its direction. In addition, if it is in a camera's sensing range, an obstacle may block the camera's FOV, which can lead to significant coverage reduction. Previous work [1], [2] deal with the coverage maximization for mobile directional sensors with only 4 orientations and does not consider obstacles. Therefore, a new approach in maximizing coverage which handles not only directional sensing range but also avoiding obstacles.

In this paper, we propose an efficient coverage maximization method for mobile video camera networks, using deep reinforcement learning. We devise two models which adjust the position of the camera node and rotate its direction, respectively. We train the two models using deep Q networks (DQN) algorithm [3]. Each camera node (i.e. DQN agent) randomly explores the environment or chooses action by its own strategy. Evaluation results show that our method enables nodes to yield up to 27% more coverage than existing methods.

*Corresponding author

II. Coverage Maximization with Mobile Cameras

In the initial stage, all sensor nodes are randomly distributed in a 2-D plane. Each sensor node starts adjusting its own position by moving itself away from other nodes until it reaches a certain distance. Once the node no longer is in range with other nodes, sensor node's FOV is adjusted so that the view coverage is maximized. For this, we develop two models: a position adjustment model (Model 1) and a FOV adjustment model (Model 2).

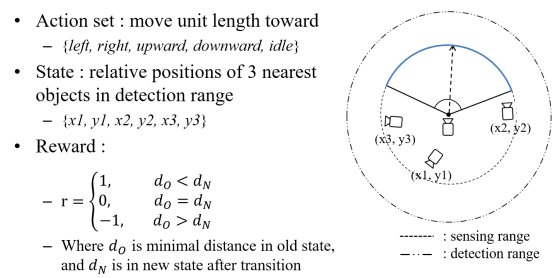


Figure 1: Description of Model 1

A. Model 1: Adjusting position

For Model 1, we leverage the virtual-force scheme in prior works [4], [5]. The objective of virtual-force scheme is to keep certain amount of distance between other sensor nodes or obstacles within its circular sensing range. We set an initial value of a proper distance to R_s, the minimal distance between any two nodes. This distance between a node to its neighboring objects gets adjusted toward coverage maximization.

In Model 1, each sensor node as a DQN learning agent detects neighboring objects in its detection range as shown in Figure 1. The state of an agent is defined as the relative coordinates of three nearest objects of its corresponding node. If the number of neighboring objects is less than 3, the state is filled up with virtual neighboring objects whose coordinate is set to (R_s, R_s). Each agent's possible action in Model 1 is to move a unit distance to left, right, upward or downward, or not to move. As a node moves, the distances to its neighboring objects and the relative coordinates of the objects change. Since a node always moves only the unit length, the amount of such changes is limited. Therefore, rather than giving a reward proportionally to the amount of change, we simply give a positive reward if a minimal distance from the neighboring objects increases; otherwise, a negative reward. If no change, the reward value is zero.

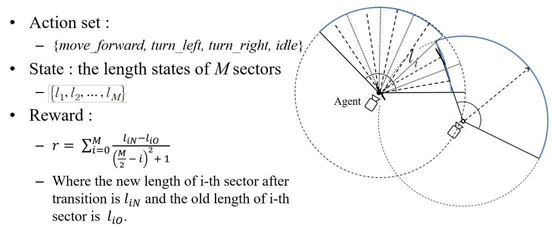


Figure 2: Description of Model 2

B. Model2 : Adjusting direction

After maintaining a distance, each node needs to change its direction to maximize its own sensing range. We consider not only coverage decreases due to neighboring objects blocking the sensing range but also overlapping with its surrounding nodes.

In model 2, each node (a DQN learning agent) divides its sensing range into M sectors and calculates an 1-dimensional size of each sector, called sector length, which represents the coverage of the sector as shown in Figure 2. A set of these sector lengths is used as the state of an agent. A sector length is obtained by calculating the distance to the sensing boundary or a neighboring node in its sensing range.

In model 2, an agent can choose 4 actions: 1) move forward, 2) rotate counter-clockwise, 3) rotate clockwise, or 4) idle. After an agent takes an action, it gets a reward from state transition. The reward function is defined as follows 1:

$$r = \sum_{i=1}^{M} \frac{l_{iN} - l_{iO}}{(M/2 - i)^2 + 1} \tag{1}$$

where l_{iN} refers to the length of the i-th sector after transition, l_{iO} refers to the length of the i-th sector before transition and M refers to the number of sectors.

Note that the denominator of Equation 1 is set to make the central part of the sensing range more valuable. This is to deal with the case when the sensing range of a node overlaps with another node's majority of its sensing area (that is, two nodes are facing each other).

III. Evaluation

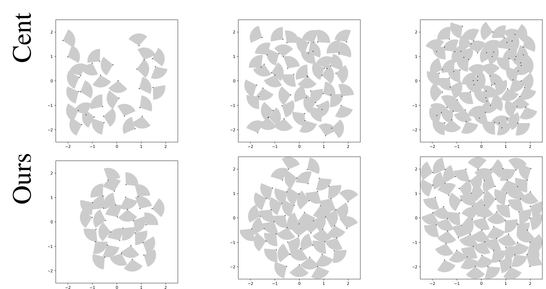
We evaluate our method by simulation using python3. We assume that a video camera has a radius of 0.02 and a viewing angle of 100 degrees and R_s of 0.5 in a 5×5 size 2D environment. M is set to 6. The unit length is 0.02 and the unit angle is 100 / 6 degrees. We train our model with the adam-optimizer in python tensorflow version 1.13.1. The discount factor is set to 0.7 and learning rate to 0.03.

We evaluate on coverage ratio: ratio of the number of the pixels covered by sensor nodes to the number of the pixels of an entire area. We compare our scheme against Cent scheme [1].

We vary the number of nodes (30, 50 and 70) and evaluate the performance of our scheme. The results are shown in Table I and the final position and coverage direction is shown in Figure 3. As shown in the figure, our method places camera nodes in a more coordinated manner in comparison to Cent scheme. Though the coverage is small in scenario with small number of nodes, nodes cover up to 4.97% more space than the previous approach. This is mainly due to the adjustment by model 2 in our method (fine grain movement & rotation) since the amount of inevitable overlapping areas becomes large in a dense condition without rotation.

	30 nodes	50 nodes	70 nodes
Cent	25.32	41.47	54.04
Ours	**25.51**	**42.42**	**58.97**

Table I: Coverage ratio (%)



(a) 30 nodes (b) 50 nodes (c) 70 nodes

Figure 3: Final status of the coverage evaluation

Acknowledgment

This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (No.2019-0-01126, Self-learning based Autonomic IoT Edge Computing) (No.2016-0-00160, Versatile Network System Architecture for Multi-dimensional Diversity)

References

[1] C.-K. Liang, M.-C. He, and C.-H. Tsai, "Movement assisted sensor deployment in directional sensor networks," in *2010 Sixth International Conference on Mobile Ad-hoc and Sensor Networks*. IEEE, 2010, pp. 226–230.

[2] M. Varposhti, P. Saleh, S. Afzal, and M. Dehghan, "Distributed area coverage in mobile directional sensor networks," in *2016 8th International Symposium on Telecommunications (IST)*. IEEE, 2016, pp. 18–23.

[3] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller, "Playing atari with deep reinforcement learning," *arXiv preprint arXiv:1312.5602*, 2013.

[4] Y. Zou and K. Chakrabarty, "Sensor deployment and target localization based on virtual forces," in *IEEE INFOCOM 2003. Twenty-second Annual Joint Conference of the IEEE Computer and Communications Societies (IEEE Cat. No. 03CH37428)*, vol. 2. IEEE, 2003, pp. 1293–1303.

[5] J. Li, L. Cui, and B. Zhang, "Self-deployment by distance and orientation control for mobile sensor networks," in *2010 International Conference on Networking, Sensing and Control (ICNSC)*. IEEE, 2010, pp. 549–553.