Online Zoom Selection Approaches for Coverage Redundancy in Visual Sensor Networks

Arezoo Vejdanparast, Peter R. Lewis, Lukas Esterle
Aston Lab for Intelligent Collectives Engineering (ALICE)
Aston University, Birmingham, UK
{vejdanpa,p.lewis,l.esterle}@aston.ac.uk

ABSTRACT

When a network of cameras with adjustable zoom lenses is tasked with object coverage, an important question is how to determine the optimal zoom level for each camera. While covering a smaller area allows for higher detection likelihood, overlapping fields of view introduce a redundancy which is vital to fault tolerance and acquisition of multiple perspectives of targets. In this paper, we study the coverage redundancy problem in visual sensor networks formalised as the k-coverage. We propose a resolution-based detection model that enables the exploration of the zoom level impact on the extension of coverage redundancy. Given mobile targets, we investigate the network-wide best k-coverage using global knowledge of the network. We explore the advantages and disadvantages of this approach and propose a realistic zoom adaptation model for cameras under a new zoom level proximity constraint. Further to the global approach, which aim to provide the highest possible k-coverage across the network, we study the performance of several online heuristics and learning approaches which use only local knowledge to approximate the highest k-coverage under our zoom level proximity constraint. Furthermore, we show that in many scenarios the dynamic behaviour resulting from the online learning approaches is not only computationally less expensive but also leads to a significant improvement on the level of coverage redundancy compared with heuristic approaches. This becomes even more apparent when objects follow scripted movement patterns.

KEYWORDS

visual sensor networks, smart cameras, online learning, machine learning approaches, decentralised systems

ACM Reference Format:
Arezoo Vejdanparast, Peter R. Lewis, Lukas Esterle. 2018. Online Zoom Selection Approaches for Coverage Redundancy in Visual Sensor Networks. In *International Conference on Distributed Smart Cameras (ICDSC '18), September 3–4, 2018, Eindhoven, Netherlands.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3243394.3243697

ICDSC '18, September 3–4, 2018, Eindhoven, Netherlands

ACM ISBN 978-1-4503-6511-6/18/09…$15.00
https://doi.org/10.1145/3243394.3243697

1 INTRODUCTION

Smart cameras are embedded devices able to perceive their environment, process this acquired knowledge, and communicate with other devices. Due to the fact that they are becoming less expensive, they are also becoming more pervasive in our environment. By operating in networks, their ability to adapt to changing conditions makes them more robust, flexible, and resilient [17, 19, 24].

In this paper we are interested in the coverage redundancy problem in networks of smart cameras with 360° fields of view (FOV). Coverage problems are concerned with how well the region of interest is monitored by a set of cameras. However, we are not trying to cover static points in the environment but rather mobile points that may change their position over time. We consider a point to be *covered* by a camera if it lies within the camera's FOV and is detectable by that camera. Furthermore, we are interested in covering points with multiple cameras. This is called k-coverage where k represents the number of cameras covering a specific point. This redundancy on the point allows for various benefits such as higher robustness of the network and multiple views on the target location. Each of our cameras is equipped with an adjustable zoom. This allows an individual camera to change the size of the sensed area. However, this capability comes with a trade-off as larger areas are perceived with lower quality and *vice versa*. This brings us to our proposed resolution-based detection model which incorporates the zoom of the camera. Applying resolution-based detection model, allows a camera to define the quality of covered objects based on their distance to the camera and its currently applied zoom. We call a specific set-up of cameras with corresponding zooms a *configuration*. This brings us to our research questions:

(1) Can we determine the best zooms for each individual camera in order to achieve the highest possible k-coverage for all available objects in the system?
(2) Is it possible for each camera to determine its own zoom, based on local information in order to achieve highest possible k-coverage for all objects in the system at all times?

To address our first research question, we identify the best zoom configuration corresponding to the highest possible value of k-coverage, across the network exhaustively using *greedy* approach. To address our second research question, We present a set of decentralised approaches to determine the zoom on the camera level without having global knowledge of the system. Furthermore, we propose a learning-based zoom selection approach using reinforcement learning algorithm namely SARSA [22] identify the best zoom for each individual camera in order to maximise the network-wide k-coverage for all objects. The rest of the paper is structured as follows. Section 2 discusses related work on the k-coverage problem. Section 3 describes the problem formally and presents the

resolution-based detection model. Next, in Section 4 we discuss the coverage approaches comprised of global approaches which aim to provide optimal k-coverage across the network using global knowledge, and online local approaches that only use local information to make decision on camera's zoom adaptation. Section 5 describes our experimental study and presents full qualitative results over all simulated scenarios. Finally, Section 6 concludes this paper and gives an outlook on our future work.

2 RELATED WORK

There is extensive research working on maximising coverage in distributed camera networks [3, 7, 15, 19, 21]. However, these works focus on minimising the overlapping area between the FOVs rather than embracing this as an opportunity to increase the redundancy in the networks.

The problem of k-coverage has been studied in several types of networks, Hefeeda et al. [13] define the k-coverage problem in dense sensor networks. The problem is to determine the minimal set of active sensors that provide required k-coverage in the network. They modeled the problem as optimal hitting set problem which is proven to be NP-hard [12]. Thus, the authors proposed a set of heuristics inspired by Brönnimann et al's. [4] approximation algorithm which can be implemented in distributed manner with low message overhead. In the context of dense networks Abrams et al. [1] studied energy efficient monitoring through partitioning the sensors into covers and activating the covers iteratively. They explored three approximation algorithms to solve the SET K-COVER problem [5] in a sensor network. The objective of these approaches are to determine a maximum number of covers where the time interval between two activation is longer. However, the aim of this work is to maintain a certain level of k-coverage all the time while in this paper we are interested in maximising the value of the minimum k for all objects. A novel approach was proposed by Alam et al. [2] to the k-coverage problem in WSNs with adjustable transmission and sensing range. The proposed solution, called on-demand k-coverage, ensures 1-coverage at the sensor deployment stage. However, in the case of event detection, the coverage will expand to k-coverage in order to increase the accuracy and robustness of covering the event. The authors discuss a trade-off between the sensor's energy consumption and the level of coverage in order to provide detection accuracy. In their work the value of k is already defined to the network and the experiments show the effectiveness of the proposed approach. However, in our work we focus on the problem of achieving the highest possible level of k-coverage across the network. Huang et al. [14] study the k-coverage problem as a decision making problem in sensor networks. The aim of the paper is to evaluate for a given k whether the sensor network is k-covered. This was studied by exploring the perimeter coverage of the sensors, considering both fixed and adjustable sensing ranges. The authors claim that the whole area is k-covered if each sensor in the network is k-perimeter covered. In their recent work Esterle and Lewis [8] investigate k-coverage on an object level. The goal was to coordinate a set of mobile cameras with a directed FOV in order for them to cover moving targets with at least k cameras. In the context of resilience to node failure Qiu et al. [18] introduce coverage hole detection and recovery approaches to achieve k-coverage in WSNs. In this work a k-coverage hole is defined as a continuous area in the region of interest including points that are covered by at most $k-1$ sensors. The first concern of the authors is how to efficiently detect the k-coverage holes in distributed way, and the second is how to avoid generating new holes while healing current holes. The first issue was tackled using a *local k-order* Voronoi diagram to help other nodes to detect holes within their own diagram. They also proposed distributed Voronoi based cooperation scheme for mobile nodes in order to tackle the second issue. Experiments show a performance improvement in terms of coverage efficiency compared to previous schemes. Further information on resilience to node failure is given in [6, 11, 16].

3 PROBLEM STATEMENT

In this work, we consider a visual sensor network (VSN) of omni-directional cameras $C = \{c_1, c_2, \ldots, c_i, \ldots, c_n\}$ with 360-degree views, each equipped with an adjustable zoom lens. The circular field of view (FOV) of camera c_i has a range r_i. The network is tasked to cover a set of moving objects $O = \{o_1, o_2, \ldots, o_j, \ldots, o_m\}$. The current zoom level has an inherent impact on the quality of the covered objects. A trade-off arises between the area of covered by the camera and the quality of acquired information. A narrower zoom covers less physical space but results in a higher pixel count for the region of interest. In contrast, a wider zoom covers more of the environment but uses fewer pixels on a given square-unit. *This may lead to objects not being detected as such even though they are within the sensing range of a camera.* While Esterle et al. [10] accounted for the deviation in quality based on the zoom, they did not incorporate the size of the object nor the distance between camera and object. In order to capture these for arbitrary objects, we consider the size of object j as the radius, q of a containing sphere, and the distance of the object from the camera as d_j. This is illustrated in Figure 1. Using this information, we can calculate the

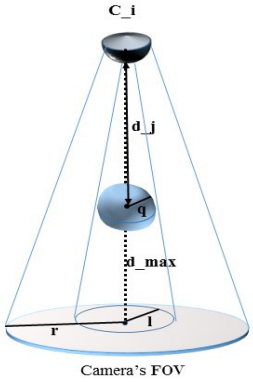

Figure 1: Demonstration of resolution-based confidence model for an object o_j inside camera c_i's FOV.

resolution in terms of the number of pixels obtained by the camera for a given object, as shown in Equation 1.

$$\frac{q}{d_j} = \frac{l}{d_{max}};\ l = \frac{q \times d_{max}}{d_j};\ \frac{l}{r} = \frac{res_j}{res_i};$$
$$res_j = \frac{res_i \times q \times d_{max}}{r \times d_j} \tag{1}$$

where d_{max} and res_i are camera-dependent parameters defining (i) the maximum distance at which a camera can still see an object with sufficient quality (visibility border), and (ii) the resolution of the camera based on the utilised sensor, respectively. This allows us to calculate the expected confidence of detection $conf_{RB}$, by determining the proportion of the overall resolution res_i used for the patch containing the projected sphere around the object. This, along with the value of constants which are provided by Esterle et al. [10], is used as follows:

$$conf_{RB}(c_i, o_j) = 0.95 \times \left(\frac{res_j}{res_i}\right) - 0.15 \tag{2}$$

An object is *covered* by a camera if that object lies within the camera's FOV and $conf_{RB}$ over that object is above a certain threshold τ. We consider d_{max} as the maximum possible value for d_j such that any object can be seen with a confidence $conf_{RB}(c_i, o_j) \geq \tau$. In general, this is dependent on q. Thus, a sensible upper limit on expected q must be chosen.

In this paper, we are not only interested in simply maximising the number of covered objects in the environment, but covering each object with as many cameras as possible. Typically in $k-$coverage problems (e.g. [1, 13, 14]), a desired fixed value of k is used, and the challenge is to ensure that all objects are covered by at least k sensors with sufficient confidence. Here, this translates to $\forall o_j$, $k_j \geq k = \sum_{i=1}^{n} \kappa_{ij}$, where

$$\kappa_{ij} = \begin{cases} 1, & \text{if } conf_{RB}(c_i, o_j) \geq \tau \\ 0, & \text{otherwise} \end{cases}$$

However, we focus on the problem of achieving the highest level of k-coverage across the network, requiring us to maximise the minimum value of k_j across all objects $o_j \in O$. We denote this minimum k value as k_{min} at time t, as defined in Equation 3. We can achieve this by adjusting the available zoom of the individual cameras and using the resolution-based sensing model defined in Equations 1 and 2.

$$k_{min}(t) = min(k_1(t), k_2(t), ...k_j(t), ..., k_m(t)) \tag{3}$$

Further, given the online nature of the problem, as objects may move about, we are also interested in maximising k_{min} over time.

Finally, a camera cannot adjust its zoom instantaneously. Zoom adjustment speed is a hardware dependent parameter, and can vary from one camera to another. Given each camera is equipped with $Z = \{z_1, z_2, ..., z_z\}$ discrete zoom levels, we model varying zoom adjustment speeds by saying that a camera can only adjust its zoom by up to δ zoom levels within a discrete time step. We call this the *zoom proximity constraint*. This results in the following time-discrete behaviour for the adaptation of zoom $z_i(t)$ of a camera c_i:

$$z_i(t+1) = z_i(t) \pm \delta. \tag{4}$$

4 COVERAGE APPROACHES

In order to maximise the value of k_{min} at any time, we employ our resolution-based model to reason about the varying abilities of cameras to cover objects within their current FOV. We propose different online camera zoom selection approaches and study their impact on the value of k_{min} over all available objects in the network. Given the dynamic nature of the problem, in which objects always move about, the question is how to select an appropriate zoom for each camera such that the value of k_{min} is maximised over time.

4.1 Greedy Approach

The *Greedy approach* analyses all potential configurations in the current time step reachable from the current zoom level. Therefore, having δ = z would allow us to select the optimal zoom configuration which provide the highest k_{min} at each time step. In this case, the highest possible network-wide k_{min} can be achieved when we are not limited by physical properties. Nevertheless, the *Greedy approach* still requires us to explore $(1+\delta)^n \leq \sigma \leq (1+2\delta)^n$ potential configurations at each time step. Where, σ is the greedy search space. However, when $\delta <$ z, the number of analysed configurations dramatically reduce and the Greedy search space will be limited to only the configurations within the boundaries of reachable zooms from current zoom.

This approach is computationally expensive and time consuming, by the time this computation completes, a moving object has probably left the FOV of the camera. This motivates us to study the performance of some online approaches that use only local information to provide near optimal outcomes.

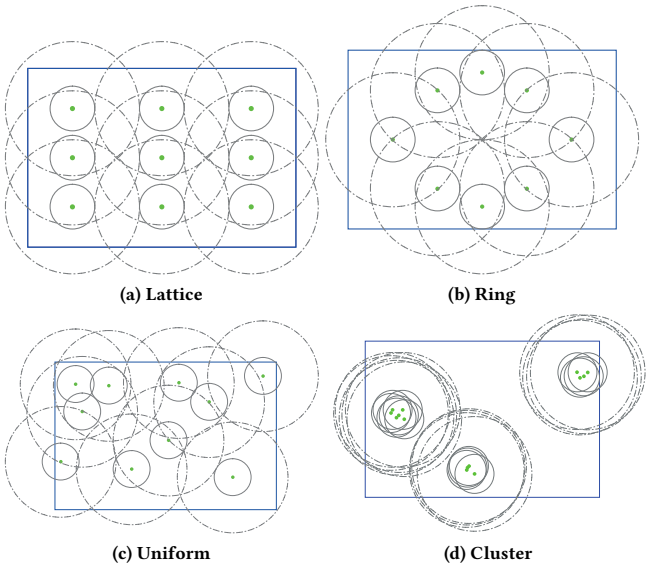

Figure 2: Camera layouts tested with the *CamSim* simulation tool. A green dot represents a camera and the associated gray inner circle demonstrate the minimum FOV when camera is zoomed in and dashed circle represent the maximum FOV associated with zoom out.

4.2 Random Approach

Our *random* approach is a simple distributed approach operating on local information alone. Each camera selects a random zoom at each time step. The zoom is sampled from a uniform distribution across all potential zooms within δ steps of the current zoom.

Table 1: Summary of Scenarios used in our Study. Lat corresponds to *Lattice Layout*, Uni is *Uniform random sampling*, Clu is *Cluster Layout*. Rand is *Random movement* of Objects while Pred is *Predefined movement*.

		#	#	Object	# Poss.	Density		
ID	Layout	Cams	Objs	Mov.	Config	1	2	3
1	Lat	9	1	Rand	1953k	9	0.015	100%
2	Uni	10	1	Rand	9765k	10	0.016	99%
3	Ring	8	1	Rand	390k	8	0.013	92%
4	Clu	14	1	Rand	6103515k	14	0.023	66%
5	Lat	9	1	Pred	1953k	9	0.015	100%
6	Uni	10	1	Pred	9765k	10	0.016	99%
7	Ring	8	1	Pred	390k	8	0.013	92%
8	Clu	14	1	Pred	6103515k	14	0.023	66%

4.3 Zoom Out Approach

In this simple and intuitive approach all cameras select the widest zoom for all time steps in order to provide the highest possible k_{min} across all available objects. This corresponds to the largest FOV fixed for all cameras throughout the simulation. It is clear that the value of δ has no impact on the performance of this approach.

4.4 Reinforcement Learning Approach

Here we apply SARSA [22] on-policy reinforcement learning method on each camera independently in order to maximise the coverage redundancy. In this approach, each camera as a learning agent estimates action-values, $Q^{\pi}(s, a)$, defined as the expected future reward associated with taking action a from state s, and thereafter, following policy π. The value of the Q function is regularly updated after every transition from state s.

In our problem, actions are defined as selecting the camera's next zoom, the action-values are the number of detected objects at that certain zoom, and finally, the cameras current state is a pair of the current zoom and number of detected objects at that zoom. In our experiments, actions are selected according to an ϵ-Greedy policy [23], in which a random action is selected with probability ϵ, and otherwise, the action with highest $Q(a)$ value is selected.

Moreover, we explored various ϵ values from 0.1, 0.01, to 0.001. In all scenarios, $\epsilon = 0.1$ obtained the closest outcomes to the optimal results. Therefore, the value of $\epsilon = 0.1$ is used for all results shown in this paper. Since standard ϵ-greedy considers all available zoom levels, this would result in δ = z. However, when $\delta <$ z, the search space of ϵ-greedy is limited to only accessible zooms including the zooms within δ-hop from the current zoom.

5 EXPERIMENTAL SIMULATION STUDY

To simulate and evaluate the performance of our approaches, we use CamSim [9], an open source distributed smart camera network simulation tool. Here, the cameras can observe their local environment with a 360° view, leading to a circular FOV where the radius represents the current zoom. In our study, the possible zoom length for an individual camera is discretised into five levels (z = 5). The distances is calculated as the 2D Euclidean distance between the camera and the location of the object. The small time slots used to calculate k_{min}, correspond to a discrete time steps which are assumed to be synchronised across all cameras in the scenario.

5.1 Test Scenarios

For the simulation framework we designed a total of 8 qualitatively different scenarios in order to evaluate the performance of our coverage approaches. We assume only one object move at a constant speed either *randomly*, where it follows a straight trajectory and bounces off the boundary of the simulation in a random direction, or along a *predefined* path. Here, our predefined path is a rectangle shaped trajectory of size $12m \times 8m$ placed in the centre of the simulation area of size $30m \times 20m$. Within the simulation environment, we consider four different camera layouts that each of which can represents a set of real world applications. An overview of these layouts is given in Figure 2. We also defined three different density metrics, to describe the properties of each scenario in more detail.

- *Density 1* is the ratio of the number of cameras to the number of objects.
- *Density 2* is the number of cameras per square unit.
- *Density 3* is the ratio of the covered area (excluding multiple overlapping FOVs) to the total simulation area.

A summary of the scenarios is provided in Table 1.

5.2 Single Object Coverage Redundancy

In this section, we evaluate the performance of our coverage approaches over covering single object following a predefined trajectory. In this regard, we conduct a set of experiments using scenarios 5-8 of Table 1.

5.2.1 Case I: No Zoom Proximity Constraint, δ = z. Here, we assume that our cameras have no physical limitations and can adjust their zoom by up to z levels within a discrete time step.

The results demonstrated in Figure 3 show that the Greedy approach under no zoom proximity constraint can provide the highest value of coverage redundancy in the network. However, since it explores all potential configurations exhaustively, it becomes a computationally expensive and time consuming approach and is not scalable for larger scenarios. The performance of other heuristics shows that our reinforcement learning algorithm outperforms the other two approaches in terms of coverage redundancy. The SARSA algorithm learns from experience to adapt the camera's zoom selection process to the object's mobility pattern by regularly updating action values after each state transition.

5.2.2 Case II: Extreme Zoom Proximity Constraint, $\delta = 1$. Here we assume the extremely limited physical properties, in which a camera is allowed to adjust its zoom to only one-hop neighbours from the current zoom. The results of the performance of our coverage approaches under $\delta = 1$ constraint are demonstrated in Figure 4. Where the object follows a predefined trajectory, applying the $\delta = 1$ constraint does not make any noticeable difference in the performance of the SARSA approach compared to δ = z. However, it leads to a significant fall in the performance of our greedy approach. This is due to restricting the search space for greedy explorations.

5.3 Impact of Different Mobility Patterns on k-Coverage

While we have already investigated the coverage redundancy obtained across single object, we are also interested in the impact of

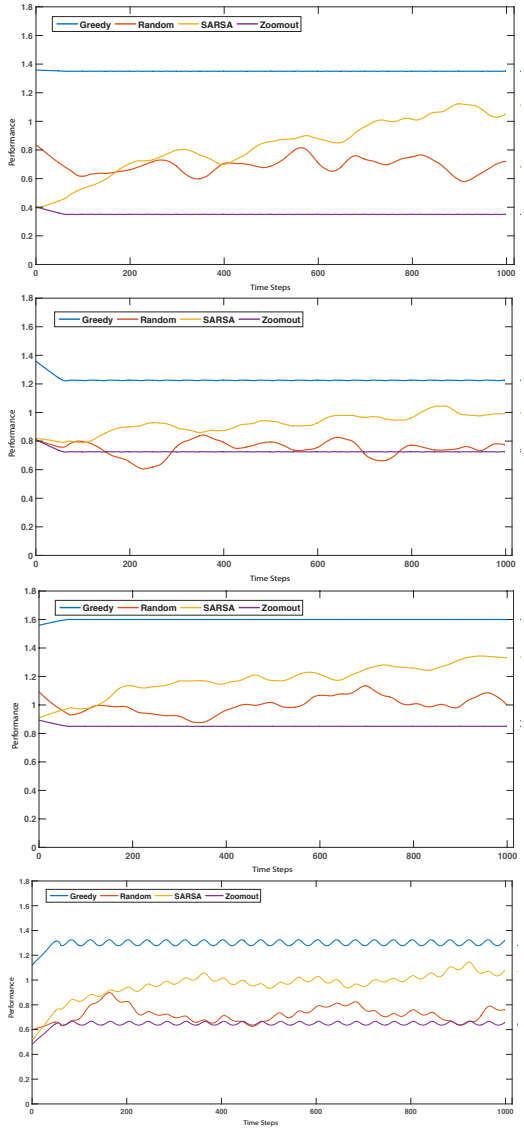

Figure 3: A selection of 4 graphs comparing performance of coverage approaches assuming δ = z using scenarios 5,6,7, and 8 (from top to bottom). The x axis of all graphs shows the simulation time steps which is 1000 and y axis shows the value of k_{min}. All results has been smoothed by locally weighted polynomial regression (LOWESS smoothing) [20]

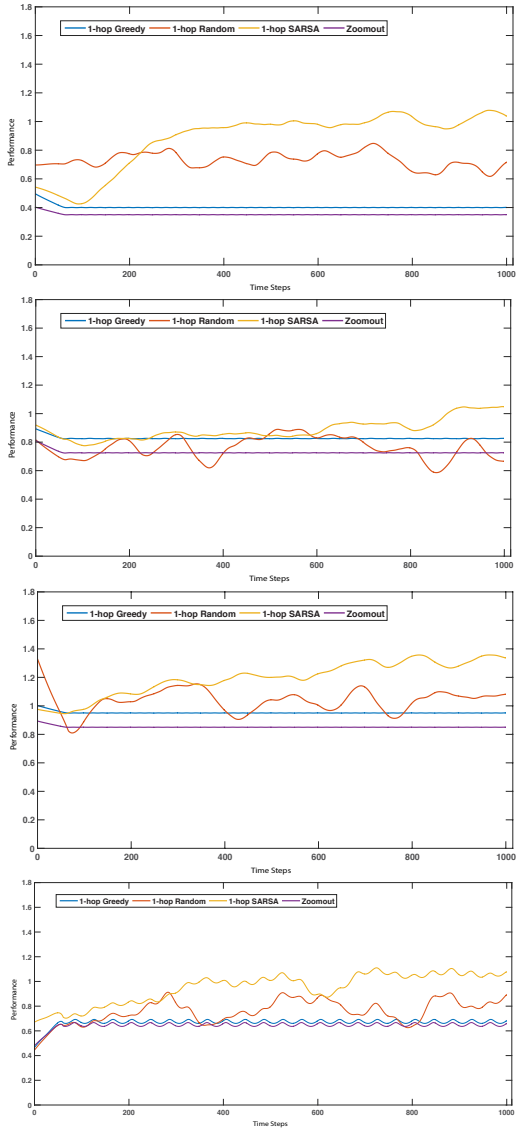

Figure 4: A selection of 4 graphs comparing performance of coverage approaches assuming δ = 1 using scenarios 5,6,7, and 8 (from top to bottom). The x axis of all graphs shows the simulation time steps which is 1000 and y axis shows the value of k_{min}. All results has been smoothed by locally weighted polynomial regression (LOWESS smoothing) [20]

different object mobility patterns on the performance of our coverage approaches. In this regard, we conducted a set of experiments across all scenarios (1-8). The first set of experiments comprises scenarios 1-4 where the object follows a straight trajectory until it reaches the environment's boundary, then bounces off in a random direction. The second set of experiments was conducted using scenarios 5-8 with scripted patterns, where objects follow a predefined rectangular path in the center of the simulation area at all times.

In order to evaluate the impact of different mobility patterns on the value of k_{min} for each approach, we used a normalised coverage performance metric for readability purposes.

$$performance = \frac{\sum_{t=1}^{t_{max}} k_{min}(t)}{\sum_{t=1}^{t_{max}} k^*_{min}(t)} \tag{5}$$

where k^*_{min} refers to the outcome of the greedy approach at each time step. We present this ratio to aid comparison. Table 2 presents the comparison across all layouts. For each layout, cells in bold show the highest performances of random and predefined movement patterns in that row.

Table 2: Comparison of the performance of different approaches using random movement (Rand) and pre-defined (PD) movement with the Greedy solution.

δ = z	SARSA		Random		Zoomout	
Layout	Rand	PD	Rand	PD	Rand	PD
Lattice	**0.68**	**0.75**	0.56	0.43	0.58	0.39
Uniform	**0.72**	**0.75**	0.54	0.51	0.61	0.59
Ring	**0.70**	**0.74**	0.51	0.47	0.60	0.53
Cluster	**0.72**	**0.74**	0.50	0.46	0.54	0.50
δ = 1	SARSA		Random		Zoomout	
Layout	Rand	PD	Rand	PD	Rand	PD
Lattice	**1.04**	**2.03**	0.94	1.70	0.92	0.94
Uniform	**1.08**	**1.17**	0.91	0.87	0.96	0.88
Ring	**1.01**	**1.21**	0.90	1.05	0.90	0.92
Cluster	**1.07**	**1.55**	0.82	1.03	0.87	0.96

5.3.1 Case I: No Zoom Proximity Constraint, δ = z. As discussed in subsection 5.2.1, the greedy approach outperforms all other approaches by providing the highest value of k_{min} across the network. The top section of Table 2 shows the fraction of the greedy approach achieved by other approaches under object's different mobility patterns. Exploring the results space, the SARSA approach outperforms the other two approaches in both random and predefined movements. The performance of SARSA itself is slightly improved with the predefined trajectory, where object follows the same trajectory all the time.

5.3.2 CASEII: Extreme Zoom Proximity Constraint, $\delta = 1$. With $\delta = 1$, the greedy search space is extremely limited. As illustrated in Figure 4, it is not necessarily the best solution. However, to aid the comparison we keep the denominator of Eq. 5 as the result of the greedy with $\delta = 1$. Therefore, the results presented in the bottom section of Table 2 show the fraction of the greedy approach that can be achieved by other approaches. The numbers that are greater than one refer to the approaches that clearly outperform the greedy approach. As clearly seen in the table, the SARSA approach outperforms all other approaches in both the predefined and the random movement.

6 CONCLUSION

We studied the problem of coverage redundancy, formalised as k-coverage in visual sensor networks. In order to explore the impact of the zoom level on the extension of coverage redundancy we proposed a resolution-based detection model. Applying the model to our cameras, leads to an important observation that simply setting all the cameras to their widest zoom does not necessarily provide the highest coverage redundancy. Further to the zoom out approach we studied the performance of greedy, random, and the reinforcement learning approaches and demonstrated the advantages and disadvantages of them on a range of scenarios using CamSim, an open source simulation tool. To model cameras with varying zoom adjustment speeds, we proposed a new problem parameter δ, where each camera can only adjust its zoom by up to δ zoom levels within a discrete time step. Under no zoom proximity constraint, the greedy approach provides the highest possible coverage redundancy across the network. However, this approach is computationally expensive and is not scalable or feasible for larger scenarios. The SARSA approach is considerably faster and provides the closest outcome to the greedy. Under extreme zoom proximity constraints, the SARSA approach provides the highest possible outcomes compared to all other approaches. It become apparent that applying even a simple learning-based approach at camera level can lead to a dynamic zoom-selection behaviour which leads to a better network-wide performance. Nevertheless, the principles behind the zoom-selection approaches and the decentralised online learning are not restricted to camera networks.

REFERENCES

[1] Z Abrams, A Goel, and S Plotkin. 2004. Set k-cover algorithms for energy efficient monitoring in wireless sensor networks. In *Proc. of Int. Conf. on Information Processing in Sensor Networks*. 424–432.

[2] K M Alam, J Kamruzzaman, G Karmakar, and M Murshed. 2014. Dynamic adjustment of sensing range for event coverage in wireless sensor networks. *Journal of Network and Computer Applications* 46 (2014), 139–153.

[3] A A Altahir, V S Asirvadam, N H B Hamid, P Sebastian, N B Saad, R B Ibrahim, and S C Dass. 2017. Optimizing Visual Surveillance Sensor Coverage Using Dynamic Programming. *IEEE Sensors Journal* 17, 11 (2017), 3398–3405.

[4] He Brönnimann and M T Goodrich. 1995. Almost optimal set covers in finite VC-dimension. *Discrete & Computational Geometry* 14, 4 (1995), 463–479.

[5] M Cardei and D-Z Du. 2005. Improving wireless sensor network lifetime through power aware organization. *Wireless Networks* 11, 3 (2005), 333–340.

[6] X Du, M Zhang, K Nygard, M Guizani, and H-H Chen. 2006. Distributed decision making algorithm for self-healing sensor networks. In *Proc. of Int. Conf. on Communications*, Vol. 8. 3402–3407.

[7] L Esterle. 2017. Centralised, Decentralised, and Self-organised Coverage Maximisation in Smart Camera Networks. In *Proc. of the Int. Conf. on Self-Adaptive and Self-Organising Systems*. 10.

[8] L Esterle and P Lewis. 2017. Online Multi-object k-coverage with Mobile Smart Cameras. In *Proc. of Int. Conf. on Distributed Smart Cameras*. 1–6.

[9] L Esterle, P R Lewis, H Caine, X Yao, and B Rinner. 2013. CamSim: A distributed smart camera network simulator. In *Proc. of Int. Conf. on Self-Adaptation and Self-Organizing Systems*. 19–20.

[10] L Esterle, B Rinner, and P R Lewis. 2015. Self-organising zooms for decentralised redundancy management in visual sensor networks. In *Proc. of Int. Conf. on Self-Adaptive and Self-Organizing Systems*. 41–50.

[11] S Galzarano, G Fortino, and A Liotta. 2012. Embedded self-healing layer for detecting and recovering sensor faults in body sensor networks. In *Proc. of Int. Conf. on Systems, Man, and Cybernetics*. 2377–2382.

[12] M R Garey and D S Johnson. 1979. A Guide to the Theory of NP-Completeness. *WH Freemann, New York* 70 (1979).

[13] M Hefeeda and M Bagheri. 2007. Randomized k-coverage algorithms for dense sensor networks. In *Proc. of Int. Conf. on Computer Communications*. 2376–2380.

[14] C-F Huang and Y-C Tseng. 2005. The coverage problem in a wireless sensor network. *Mobile Networks and Applications* 10, 4 (2005), 519–528.

[15] K R Konda, N Conci, and F De Natale. 2016. Global Coverage Maximization in PTZ-Camera Networks Based on Visual Quality Assessment. *IEEE Sensors Journal* 16, 16 (2016), 6317–6332.

[16] L Massoulié, A-M Kermarrec, and A J Ganesh. 2003. Network awareness and failure resilience in self-organizing overlay networks. In *Proc. of Int. Symp. Reliable Distributed Systems*. 47–55.

[17] C Piciarelli, L Esterle, A Khan, B Rinner, and G L Foresti. 2016. Dynamic Reconfiguration in Camera Networks: A Short Survey. *IEEE Transactions on Circuits and Systems for Video Technology* 26, 5 (2016), 965–977.

[18] C Qiu, H Shen, and K Chen. 2015. An energy-efficient and distributed cooperation mechanism for k-coverage hole detection and healing in WSNs. In *Proc. of Int. Conf. on Mobile Ad Hoc and Sensor Systems*. 73–81.

[19] B Rinner, B Dieber, L Esterle, P R Lewis, and X Yao. 2012. Resource-aware configuration in smart camera networks. In *Proc. of Int. Conf. on Computer Vision and Pattern Recognition*. 58–65.

[20] P Royston. 1992. Lowess smoothing. *Stata technical bulletin* 1, 3 (1992).

[21] S Rudolph, S Edenhofer, S Tomforde, and J Hähner. 2014. Reinforcement Learning for Coverage Optimization Through PTZ Camera Alignment in Highly Dynamic Environments. In *Proc. of the Int. Conf. on Distributed Smart Cameras*. 1–6.

[22] R S Sutton and A G Barto. 1998. *Introduction to reinforcement learning*. Vol. 135. MIT press Cambridge.

[23] J Vermorel and M Mohri. 2005. Multi-armed bandit algorithms and empirical evaluation. In *ECML*, Vol. 3720. 437–448.

[24] W Wolf, B Ozer, and T Lv. 2002. Smart cameras as embedded systems. *Computer* 35, 9 (2002), 48–53.