IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received September 10, 2019, accepted October 2, 2019, date of publication October 14, 2019, date of current version November 7, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2947262

A Distributed Barrier Coverage Mechanism for Supporting Full View in Wireless Visual Sensor Networks

PEI XU[1], I-HSIUNG CHANG[2], CHIH-YUNG CHANG[2], (Member, IEEE), BHARGAVI DANDE[2], AND CHIH-YAO HSIAO[2]

[1]School of Computer Science and Technology, Anhui University, Hefei 230601, China
[2]Department of Computer Science and Information Engineering, Tamkang University, Taipei 25137, Taiwan

Corresponding author: Chih-Yung Chang (cychang@mail.tku.edu.tw)

This work was supported by the Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-032-015-MY3 and Grant MOST 108-2622-E-032-002-CC3.

ABSTRACT Barrier coverage is one of the most important issues in intruder detection applications. Visual sensors can give more accurate information of intruder but also raise the problem of large data volume for information exchange and processing. Constructing a disjoint *full-view* barrier using minimal number of visual sensors has been the main goal for handling the barrier coverage problem. This paper presents a decentralized FBCA mechanism which consists of *Region Partitioning Phase*, *Grid Excluding Phase*, *Grid Verification Phase* and *Full-View Barrier Construction Phase*. In *Region Partitioning Phase*, the given *R* is partitioned into a set of equal-sized grids, aiming to simplify the construction problem of the full-view barrier. In *Grid Excluding Phase*, a certain amount of grids is removed, aiming to reduce the computational complexity. In *Grid Verification Phase*, each visual sensor aims to check if its neighboring grids satisfy the full-view coverage criteria. Finally, the *Full-View Barrier Construction Phase* aims to construct as many as possible full-view barriers. Experimental study shows that the proposed FBCA outperforms existing work SP, and likely approaches to the optimal solution (MCSPS) in terms of the average success ratio for constructing a *full-view* barrier.

INDEX TERMS Intruder detection, surveillance, *full-view* barrier, wireless visual sensor networks (WVSNs).

I. INTRODUCTION

Wireless visual sensor networks (WVSNs) consist of a number of visual sensors. The visual sensors have been widely used for many applications, including image monitoring service [1], [2], objects location identification [3] as well as constructing a virtual fence [4], [5] on the border between Mexico and the United States. The k-barrier coverage problem [6]–[10] has been widely discussed for wireless sensor networks (WSNs) in the past few years. A barrier constructed by a set of sensors in a given monitored region R is said to satisfy k-barrier coverage if any crossing path is detected by at least k distinct sensors [6], where a path is said to be a crossing path if the path crosses through the complete width of R. Compared with the traditional sensor in WSNs, each visual sensor in WVSNs mounts a camera which provides rich information such as image for better understanding the behaviors of targets in target tracking applications.

Different from the traditional sensor, the sensing range of a visual sensor is a Field of View (FoV), which can be viewed as a fan-shaped. Hence the existing barrier-coverage algorithms [6]–[10] developed for WSNs cannot be applied to the WVSNs. In the past few years, the barrier coverage problem in WVSNs has attracted much attention. Since the visual sensors provide rich information which requires computation cost, most of the existing approaches [11]–[13] aim to reduce the number of active visual sensors. However, they did not guarantee that every perspective of an object at any position is under the view of some visual sensors. Study [14] proposed the required conditions for satisfying the full-view coverage while study [15] proposed a method for constructing a barrier that satisfies the full-view coverage. As an extension of study [15], study [16] further proposed a method aiming to minimize the number of visual sensors for constructing

The associate editor coordinating the review of this manuscript and approving it for publication was Hongwei Du.

the *full-view* barrier. However, according to our knowledge, most existing studies which considered the issue of full-view coverage have high computation cost and requires high communication cost since they are centralized.

This paper addresses the problem of full-view barrier coverage and develops a decentralized Full-View Barrier Construction Algorithm (FBCA) for WVSNs. The developed FBCA aims to construct a maximal number of barriers. Each constructed barrier is composed of as few visual sensors as possible while the full-view coverage requirement can be guaranteed. Consequently, those visual sensors that participate in the constructed full-view barriers can work in turns and the purpose of load balance can be achieved.

The following presents the main contributions of this paper.

1) **Decentralized solution for the construction of the full-view barrier.** By locally considering the contribution of each visual sensor, the proposed FBCA can construct the barrier which satisfies the full-view coverage requirement in a decentralized manner.
2) **Similar performance as compared with the optimal solution.** The FBCA can achieve similar performance as compared with the optimal solution [16], in terms of the number of required sensors.
3) **Fewer required sensors for achieving full view.** Compared with [15], the proposed FBCA requires fewer sensors for constructing the full view barrier.
4) **Low computation and communication costs.** The FBCA partitions the monitoring region into a number of grids. The grid-based approach reduces both the computational complexity and the number of control overheads.

The remainder of this paper is organized as follows. Section II discusses the related works. Section III introduces the network environment and assumptions. Section IV gives a detailed description of how to select a set of visual sensors for constructing the full-view barrier. Section V presents the simulation results while Section VI concludes this paper.

II. RELATED WORK

This section presents the related works of barrier coverage developed for WSNs and WVSNs. Then, the works related to full-view barrier are introduced.

A. BARRIER COVERAGE IN WSNs

The barrier coverage problem for traditional WSNs has been studied extensively in the past few years. Kumar *et al.* [6] firstly defined the notion of k-barrier coverage of a belt region for wireless sensors. They proposed an algorithm to determine whether a belt region supports the k-barrier coverage. Kumar *et al.* [7] further proposed a centralized sleep-wake-up schedule, aiming to prolong the lifetime of wireless sensor networks. Chen *et al.* [8] introduced the concept of local barrier coverage for WSNs. They proposed a sleep-wake up algorithm, aiming to maximize the network lifetime. Huang and Tseng [9] proposed a centralized algorithm for constructing the barrier for WSNs. In the proposed algorithm, the boundary arcs of sensing intersection area among sensors are calculated. A set of sensors with large projection length will be selected to form the barrier.

Yang and Qiao [10] investigated the effectiveness of sensing range of each sensor on the barrier and derived the projection length of each sensor. A centralized scheme which combines all the projection lengths was developed for constructing the barrier. However, the studies [6]–[10] mentioned above were developed for scalar sensors. They cannot be efficiently applied to WVSNs because the sensing area of each visual sensor is fan-shaped, which is completely different from a circle-shaped sensing area of the scalar sensor.

B. BARRIER COVERAGE IN WVSNs

The barrier coverage problem for WVSNs has gradually received much attention. Ghazalian *et al.* [11] analyzed deployment strategies with FoV sensing model of visual sensor. They proposed connectivity checking and repairing methods, aiming to maintain the connectivity between visual sensors. However, the authors did not consider the construction of a defense barrier. Yang *et al.* [12] introduced the concept of weighted barrier graph (WBG) aiming to model the barrier coverage problem. They demonstrated that the problem of finding the number of visual sensors is equivalent to the problem of constructing the shortest path from the source node to the destination node on the *WBG*. Based on the Hungarian algorithm, an approach was proposed aiming to explore all the paths on WBG.

As an extension of study [12], Wang *et al.* [13] further proved that the construction of a barrier with k-coverage is equivalent to finding k vertex-disjoint paths with the minimum total length on the WBG. Then, an algorithm based on previous work, called Vertex-Disjoint Path approach, was proposed for solving the problem of k-barrier coverage. However, studies [11]–[13] did not consider the full-view requirement.

C. FULL-VIEW BARRIER COVERAGE IN WVSNs

In literature, study [14] first defined the full-view coverage. Furthermore, a full-view verification method was proposed in [14], aiming to verify whether or not a given monitored region is satisfied with the cover requirement of full-view coverage. Then, an estimation of deployment density was presented to achieve full-view coverage for a given monitored region. As an extension of study [14], Wang and Cao [15] further studied the construction of the full-view barrier. The monitored field is partitioned into a number of sub-regions. The sub-regions are transformed into a graph composed of vertices and edges. Each vertex denotes a sub-region and an edge denotes a connected relationship between two neighboring sub-regions. Based on the graph, a method is proposed to verify whether or not each vertex satisfies the full-view coverage. A path, which is composed of full-view vertices, will be discovered from leftmost to rightmost boundaries on the graph by applying Dijkstra's algorithm. This path represents a set of contiguous full-view sub-regions across the field, which is essential for constructing the full-view barrier.

As an extension of study [15], another study [16] further proposed a method, aiming to minimize the number of visual sensors for constructing the *full-view* barrier. The method firstly partitioned the monitoring region into several disjoint sub-regions. The sub-regions are then classified into full-view-covered regions or not-full-view covered regions. After that, a weighted graph is built according to the full-view-covered regions and their relationships. Based on the graph, an optimal algorithm is proposed to select the minimum number of visual sensors for constructing the full-view barrier. Though both studies [15] and [16] can construct a barrier satisfying the full-view requirement, they have high computational cost and require high communication cost because they are centralized. In addition, the number of constructed full-view barriers can be further improved. In this paper, a decentralized FBCA is proposed aiming to construct a satisfactory number of full-view barriers.

III. NETWORK ENVIRONMENT AND PROBLEM FORMULATIONS

This section initially introduces the network environment and assumptions of the considered WVSNs. Then, the problem formulations of this work are proposed.

A. NETWORK ENVIRONMENT

The considered monitoring region is a rectangle region R with size $L \times W$, where L and W are the length and width of R, respectively. Let notations b_{top}, b_{bottom}, b_{left}, and b_{right} denote the top, bottom, left, and right boundaries of R, respectively. A path is said to be a *crossing path* if it completely crosses from b_{bottom} to b_{top}. A set of n visual sensors $S = \{s_1, s_2, \ldots, s_n\}$ are randomly deployed in R. Each visual sensor s_i has a unique ID and is aware of its own location and the boundaries of R. As shown in Fig.1, each s_i is fixed and has a non-rotatable *sensing range* with sensing radius r, communication radius $2r$, Field-of-View (FoV) angle θ, and orientation vector $\overrightarrow{f_i}$. Let $L(s_i)$ denote the visual sensor's position and the energy required for each s_i is assumed to be sufficiently supported. This can be achieved by many existing Energy Harvesting technologies [17]. Through the exchange of beacon messages with one-hop neighbors, each s_i can collect IDs, locations, and information of sensing range of its neighboring visual sensors.

As shown in Fig. 1, let vector $\overrightarrow{v}$ denote the facing direction (e.g. the facing direction of intruder). Let $\alpha\left(\overrightarrow{v_1}, \overrightarrow{v_2}\right)$ denote the angle between vectors $\overrightarrow{v_1}$ and $\overrightarrow{v_2}$, where $0 \leq \alpha\left(\overrightarrow{v_1}, \overrightarrow{v_2}\right) \leq \pi$. A point p is said to be covered by s_i if p is in the sensing range of s_i. That is, $|s_ip| \leq r$ and $\alpha\left(\overrightarrow{s_ip}, \overrightarrow{v}\right) \leq \varphi$, where $|s_ip|$ denotes the Euclidean distance between s_i and point p and $\overrightarrow{s_ip}$ denotes the vector from s_i to p. Some definitions are introduced as follows, which will be utilized throughout the paper.

Definition 1 (Full-View Coverage): A point p is *Full-View Covered* if for any facing direction $\overrightarrow{v}$, there always exists a visual sensor s_i, such that p is covered by s_i and $\alpha\left(\overrightarrow{s_ip}, \overrightarrow{v}\right) \leq \varphi$, where $\varphi\ (\in [0, \pi/2))$ is a predefined parameter and is called the *effective viewing angle*.

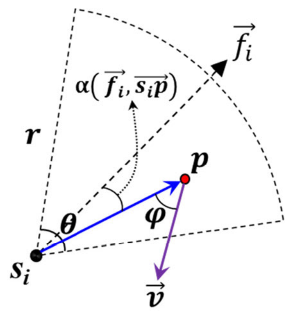


FIGURE 1. The sensing model and the full-view coverage model.

Definition 2 (Full-View Barrier): Given a region R, a barrier constructed by a set of visual sensors is said to have the *full-view* capability if any point in the crossing path from the boundary b_{bottom} to the boundary b_{top} is full-view covered.

B. PROBLEM FORMULATIONS

Let N_A denote the number of disjoint full-view barriers constructed in R by applying a certain algorithm A. As shown in Exp. (1), the objective function of this paper aims at developing an algorithm A for constructing the maximum number of disjoint full-view barriers.

$$\textbf{maximize}\ (N_A) \tag{1}$$

Exp. (2) also indicates that the number of visual sensors composing each full-view barrier should be minimized in order to maximize the number of constructed disjoint full-view barriers. This goal can help reduce the complexities of intrusive information, computing, and communication. Since the constructed full-view barriers can work in turns, the visual sensors can be maintained in a better way and their workloads are balanced.

A region is said to be the *full-view region* if any point in it satisfies the full-view coverage. Let $reg_{\hat{j}}$ denote the j^{th} *full-view region*. Let $S(reg_{\hat{j}})$ denote the *set of visual sensors* that can cooperatively provide the capability of full-view coverage for reg_{j}. Let $FB_h = \left(S(reg_{\hat{1}}), S(reg_{\hat{2}}), \ldots, S(reg_{\hat{f_h}})\right)$ denote the h^{th} constructed full-view barrier which is composed of $\hat{f_h}$ ordered *sets of visual sensors*. Constraint (2) asks that any two of the neighboring full-view regions in FB_h should be intersected.

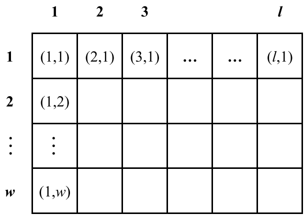
Continuous constraint:

$$reg_{\hat{j}} \cap reg_{\hat{f}+1} \neq \emptyset, \quad 1 \leq \hat{j} \leq \hat{f_h} - 1,\ \forall reg_{\hat{j}}, reg_{\hat{f_h}} \in FB_h \tag{2}$$

Another constraint called *boundary constraint*, should be satisfied for the *first full-view region* $reg_{\hat{1}}$ and *last full-view region* $reg_{\hat{f_h}}$ in FB_h. The Exp. (3) shows that the $reg_{\hat{1}}$ and $reg_{\hat{f_h}}$ should be intersected with the b_{left} and b_{right}, respectively.

Boundary constraint:

$$reg_{\hat{1}} \cap b_{left} \neq \emptyset \wedge reg_{\hat{f_h}} \cap b_{right} \neq \emptyset, \quad \forall reg_{\hat{1}}, reg_{\hat{f_h}} \in FB_h \tag{3}$$



	1	2	3			*l*
1	(1,1)	(2,1)	(3,1)	…	…	(*l*,1)
2	(1,2)					
⋮	⋮					
w	(1,*w*)					

FIGURE 2. Grid-based partition.

Let $x_{i,j}^{h}$ denote the *Boolean* value indicating whether or not the visual sensor $s_i \in S(reg_{\hat{j}})$ participates in FB_h. That is,

$$x_{i,j}^{h} = \begin{cases} 1, & \text{if } s_i \in S\left(reg_{\hat{j}}\right), reg_{\hat{j}} \in FB_h \\ 0, & \text{otherwise.} \end{cases} \tag{4}$$

Constraint (5) validates the participation of each sensor s_i in at most one full-view barrier. In other words, any two full-view barriers like FB_α and FB_β should be *sensor-disjoint*.

Sensor-disjoint constraint:

$$\sum_{\forall FB_h} x_{i,j}^{h} \leq 1, \quad \forall i \in S \tag{5}$$

The necessity of *sensor-disjoint* is explained as follows. For instance, consider any two constructed full-view barriers, says FB_α and FB_β. The main working function of two full-view barriers is to achieve the purpose of load balance. However, when there exists a common visual sensor participating in both the FB_α and FB_β, it needs to work all the time and hence the purpose of load balance cannot be achieved.

IV. FULL-VIEW BARRIER CONSTRUCTION ALGORITHM (FBCA)

The proposed FBCA mainly consists of four phases, including the *Region Partitioning Phase*, *Grid Excluding Phase*, *Grid Verification Phase*, and *Full-View Barrier Construction Phase*. In *Region Partitioning Phase*, the given region R is partitioned into a set of equal-sized grids, aiming to simplify the construction problem of full-view barrier. In *Grid Excluding Phase*, a certain amount of grids will be removed, aiming to reduce the computational complexity. In *Grid Verification Phase*, each visual sensor aims to check if its neighboring grids satisfying the full-view coverage. Finally, the *Full-View Barrier Construction Phase* has two major tasks. The first one aims to transfer the relationships among those grids satisfying the full-view coverage to a local *weighted graph*. Based on the *weighted graph*, the second task further constructs the full-view barrier in a decentralized manner. The following presents the detailed operations designed in each phase.

A. REGION PARTITIONING PHASE

In the region partitioning phase, each visual sensor s_i has two major tasks to be accomplished. The first task is *region partitioning* which aims to partition the given region R into a set of equal-sized grids, as shown in Fig. 2. The second task of each s_i is the *coverage identification task* which aims to identify the coverage degree of the grid it located. Let the size of the grid be $l \times l$. In the first task, each grid is assigned with coordinates (x, y) which are denoted by $g_{x,y}$. As shown in Fig. 2, the most top-left grid is assigned with coordinates (1, 1) and the x-coordinate and y-coordinate are increased by one if the location of a grid shifts one position towards the right and down directions, respectively. Each s_i in this phase will firstly identify the coordinates of the grid it located.

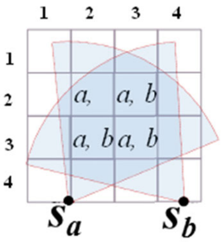
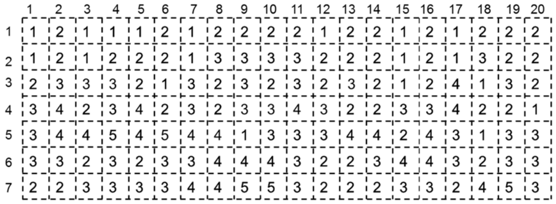


FIGURE 3. The fully covered grids of s_a and s_b.



	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
1	1	2	1	1	1	2	1	2	2	2	2	1	2	2	1	2	1	2	2	2
2	1	2	1	2	2	2	1	3	3	3	3	2	2	2	1	2	1	3	2	2
3	2	3	3	3	2	1	3	2	3	2	3	2	3	2	1	2	4	1	3	2
4	3	4	2	3	4	2	3	2	3	3	4	3	2	2	3	3	4	2	2	1
5	3	4	4	5	4	5	4	4	1	3	3	3	4	4	2	4	3	1	3	3
6	3	3	2	3	2	3	3	4	4	4	3	2	2	3	4	4	3	2	3	3
7	2	2	3	3	3	3	4	4	5	5	3	2	2	2	3	3	2	4	5	3

FIGURE 4. Weighted grid matrix (WGM).

In the second task, let the term *coverage degree* denote the number of visual sensors that can fully cover the specific grid. For example, the grid $g_{2,2}$ in Fig. 3 is only fully covered by the sensor s_a. Thus, the coverage degree of $g_{2,2}$ is 1. Besides, since the $g_{3,3}$ is fully covered by both sensors s_a and s_b, its coverage degree is 2. Each s_i in this task will evaluate the *coverage degrees* of those grids that are fully covered by itself, by exchanging the beacon messages with its neighbors. As a result, the *Weighted Grid Matrix* (WGM) shown in Fig. 4 can be established by marking the coverage degrees on the corresponding grids. Herein, we emphasize that the developed WGM is a tool, which is used to help understand the proposed *FBCA* from the conceptual point of view. The developed *FBCA* is totally decentralized.

A grid is said to be a *full-view grid* if it satisfies the full-view coverage. In concept level, the full-view barrier can be constructed by the composition of a series of connected full-view grids from boundaries b_{left} to b_{right} in *WGM*. Therefore, the *Grid Verification Phase* aims to verify whether or not a given grid in *WGM* is the full-view grid. Before that, in the next phase, called *Grid Excluding Phase*, a number of grids will not be considered to be verified. The following presents the details of the *Grid Excluding Phase.*

B. GRID EXCLUDING PHASE

To reduce the computation and communication costs of the next phase, this phase aims to remove some unqualified grids.

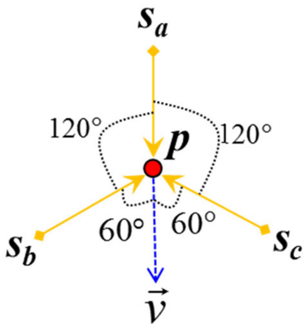


FIGURE 5. Weighted grid matrix (WGM).

According to the description given in **Definition 1**, a grid $g_{x,y}$ is said to be a full-view grid if every point $p \in g_{x,y}$ is full-view covered. That is, the gird $g_{x,y}$ can be ignored if there exists a point $p \in g_{x,y}$ that is not full-view covered. Recall that a point p is full-view covered, if for any facing direction $\vec{v}$ of intruder located on point p, there always exists a visual sensor s_i, such that p is satisfied with two conditions: *1)* covered by s_i and *2)* $\alpha\left(\overrightarrow{s_ip}, \vec{v}\right) \leq \varphi$. The following describes how to ignore the disqualified grids.

Excluding Condition 1 (Coverage Condition):

Grid $g_{x,y}$ does not hold the first condition if it is with zero *coverage degree*. □

The zero *coverage degree* represents that no visual sensor can cover any point $p \in g_{x,y}$. Therefore, grid $g_{x,y}$ with zero *coverage degree* can be ignored in the later phase. The following further describes the use of the second condition. Consider the grid $g_{x,y}$. Let point p be the central point of $g_{x,y}$. The following proposes the second excluding condition for ignoring grid $g_{x,y}$.

Excluding Condition 2 (Minimal Number Condition):

Grid $g_{x,y}$ can be ignored if the number of sensors cover point p is smaller than N^{min}. Equ. (6) calculates the value of N^{min}.

$$N^{min} = \frac{2\pi}{2\varphi}, \quad \varphi \in [0, \pi/2) \tag{6}$$

It is easy to evaluate the minimal number of visual sensors that cover the grid $g_{x,y}$. Since each visual sensor contribute to point p at most 2φ coverage angle, point p is the full-view point if the number of visual sensors that covers p at least $2\pi/2\varphi$. Fig. 5 gives an example of applying the Minimal Number Condition. Assume that the *effective viewing angle* φ is 60°. This also indicates that each visual sensor will contribute to monitoring point p at most 120°. Therefore, the value of N^{min} is 3. Consequently, as shown in Fig. 4, those grids whose coverage degrees less than three can be ignored in this phase.

Even though the grid $g_{x,y}$ does not satisfy the two conditions given above, it is still possible that $g_{x,y}$ is a full-view grid. The next phase, called Grid Verification Phase help verify whether or not grid $g_{x,y}$ is a full-view grid.

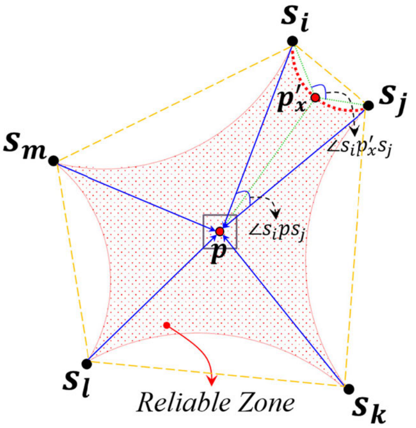


FIGURE 6. Construction of *reliable zone*.

C. GRID VERIFICATION PHASE

Let $S(g_{x,y})$ denote the set of visual sensors that can fully cover grid $g_{x,y}$. For any point $p \in g_{x,y}$, a *vector set* $VS = \left\{\overrightarrow{s_1p}, \overrightarrow{s_2p}, \ldots, \overrightarrow{s_m\,p}\right\}$ is a collection of every vector from each visual sensor $s_i \in S(g_{x,y})$ to point p.

Point Full-View Condition: A point p is full-view covered if and only if the rotation angle from $\overrightarrow{s_ip}$ to $\overrightarrow{s_{(i+1) mod\ m}\,p}$ is less than or equal to 2φ, where $1 \leq \hat{i} \leq m$, $\overrightarrow{s_ip} \in VS$.

Applying the *point full-view condition* to determine whether or not the grid $g_{x,y}$ is a full-view grid requires the tasks that every point $p \in g_{x,y}$ needs to be verified. This will result in high computation and communication costs. In the following, we introduce a concept of the *reliable zone* that is defined as the zone expanded from a full-view point p. Every grid inside the *reliable zone* will be guaranteed to be the full-view grid. The following presents a method, aiming to construct the *reliable zone*. Let $\angle s_ips_j$ denote the angle between $\overrightarrow{s_ip}$ and $\overrightarrow{s_jp}$, where $\overrightarrow{s_ip}$ and $\overrightarrow{s_jp}$ belong to VS. Let Δs_ips_j denote the *triangle area* constructed by connecting points s_i, p, snd s_j. As shown in Fig. 6, assume that point p is a *full-view* point. According to previously proposed *Point Full-View Condition*, condition $\angle s_ips_j \leq 2\varphi$ holds. That is, point p toward the range of angles between $\overrightarrow{s_ip}$ and $\overrightarrow{s_jp}$ would be face detected by s_i or s_j. There must exists a set of points, denoted by p'_u, which satisfies the condition $\angle s_ip'_xs_j = 2\varphi$. Let *Boundary Point Set* S^{BPoint} be the set of collection of all points $p'_u \in \Delta s_ips_j$, as presented in (7).

$$S^{BPoint} = \left\{p'_x \mid \angle s_ip'_us_j = 2\varphi, p'_u \in \Delta s_ips_j\right\} \tag{7}$$

As shown in Fig. 6, the positions of s_i, p, s_j, and $\forall p'_u \in S^{BPoint}$ constitute a *safe region*. The *safe region* guarantees that any facing direction from any point inside it toward the range of angles between $\overrightarrow{s_ip}$ and $\overrightarrow{s_jp}$ would be face detected by visual sensors s_i or s_j. This occurs because that the angle between $\overrightarrow{s_ip'_u}$ and $\overrightarrow{s_jp'_u}$ will always be less than and equal to 2φ for any point p'_u inside the *safe region*. Similarly, every triangle area, including Δs_ips_j, Δs_jps_k, Δs_kps_l, Δs_lps_m, and Δs_mps_i, exists its own *safe region*. The union of these *safe regions* will constitute the *reliable zone*. The following

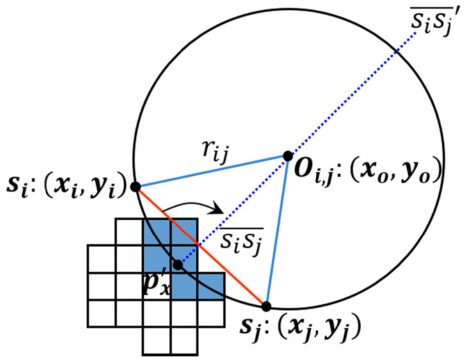


FIGURE 7. Construction of *safe region*.

describes the construction of *safe region*, which is the major part for constituting the *reliable zone*.

Herein, we note that, for any type of triangle area, there always exists a unique *circumscribed circle*. As shown in Fig. 7, assume that point $p_u' \in S^{BPoint}$. That is, condition $\angle s_i p_u' s_j = 2\varphi$ is satisfied. Since there exists a unique *circumscribed circle* composed of triangle area $\Delta s_i p_u' s_j$, the three vertices of triangle area $\Delta s_i p_u' s_j$ intersect with the fixed arc from s_i to s_j. Let $\widehat{s_i p_u' s_j}$ denote the arc passing through the three points s_i, p_u', and s_j. Let $\overline{s_i p}$ denote the line connected from position of s_i to point p. The *safe region* enclosed by lines $\overline{s_i p}$ and $\overline{s_j p}$ and the arc $\widehat{s_i p_u' s_j}$, denoted by $\Psi s_i p s_j p_u'$, guarantees that any facing direction from any point inside it toward the range of angles between $\overrightarrow{s_i p}$ and $\overrightarrow{s_j p}$ would be face detected by visual sensors s_i or s_j. This fact occurs because that moving p_u' toward the position of point p will cause the condition $\angle s_i p_u' s_j < 2\varphi$. Since the *safe region* $\Psi s_i p s_j p_u'$ can be constructed if the arc $\widehat{s_i p_u' s_j}$ is derived, the following presents how to construct the *circumscribed circle*. As soon as the *circumscribed circle* is identified, the arc $\widehat{s_i p_u' s_j}$ is also identified accordingly.

As shown in Fig. 7, let C_{ij} denote the *circumscribed circle* intersected with the points s_i, p_u', and s_j. Let r_{ij} and O_{ij} denote the radius and center of C_{ij}, respectively. To determine the *circumscribed circle*, both the r_{ij} and O_{ij} needs to be known. The derivations of them are illustrated as follows.

1) DERIVATION OF r_{ij}

Let (x_i, y_i) denote the coordinates of location of sensor s_i and $\left(x_u^p, y_u^p\right)$ denote the coordinates of location of point p_u'. The size of triangle area $\Delta s_i p_u' s_j$ can be evaluated by applying Equ. (8).

$$\Delta s_i p_x' s_j = \frac{1}{2} \cdot \begin{vmatrix} x_i & x_u^p & x_j & x_i \\ y_i & y_u^p & y_j & y_i \end{vmatrix} \tag{8}$$

Let $\left|\overline{s_i s_j}\right|$ denote the Euclidean distance between sensors s_i and s_j. Let $\left|\overline{s_i p_u'}\right|$ denote the Euclidean distance between positions of s_i and point p_u'. The length of each side of triangle area $\Delta s_i p_u' s_j$, including the $\left|\overline{s_i s_j}\right|$, $\left|\overline{s_i p_u'}\right|$, and $\left|\overline{s_j p_u'}\right|$, are evaluated

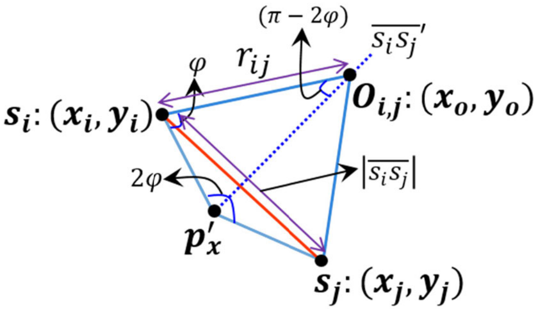


FIGURE 8. Evaluation of $O_{i,j}$.

by applying Equ. (9)

$$\begin{aligned} \left|\overline{s_i s_j}\right| &= \sqrt{\left(x_j - x_i\right)^2 + \left(y_j - y_i\right)^2} \\ \left|\overline{s_i p_x'}\right| &= \sqrt{(x_j - x_u^p)^2 + (y_j - y_u^p)^2} \\ \left|\overline{s_i p_x'}\right| &= \sqrt{(x_j - x_u^p)^2 + (y_j - y_u^p)^2} \end{aligned} \tag{9}$$

According to Equs. (8) and (9), the r_{ij} can be evaluated by applying Equ. (10).

$$r_{ij} = \frac{\left|\overline{s_i s_j}\right| \cdot \left|\overline{s_i p_u'}\right| \cdot \left|\overline{s_j p_u'}\right|}{4 \cdot \Delta s_i p_u' s_j} \tag{10}$$

2) DERIVATION OF O_{ij}

Let $\widetilde{s_i s_j}$ denote the perpendicular bisector of $\overline{s_i s_j}$ and $\widetilde{s_i p_u'}$ denote the perpendicular bisector of $\overline{s_i p_u'}$. Since O_{ij} is the circumcenter of triangle $\Delta s_i p_u' s_j$, it can be derived by finding the intersection point between $\widetilde{s_i s_j}$ and $\widetilde{s_i p_u'}$. The equations of $\widetilde{s_i s_j}$ and $\widetilde{s_i p_u'}$ can be evaluated by applying Equs. (11) and (12), respectively.

$$\left(x_i - x_j\right) x + \left(y_i - y_j\right) y = \frac{(x_i)^2 + (y_i)^2 - \left(x_j\right)^2 - \left(y_j\right)^2}{2} \tag{11}$$

$$\left(x_i - x_u^p\right) x + \left(y_i - y_u^p\right) y = \frac{(x_i)^2 + (y_i)^2 - \left(x_u^p\right)^2 - \left(y_u^p\right)^2}{2} \tag{12}$$

Let (x_o, y_o) denote the coordinates of location of O_{ij}.The (x_o, y_o) can be evaluated by substituting Equs. (11) and (12). That is,

$$\begin{aligned} x_o &= \frac{\left(x_u^p\right)^2 - \left(x_j\right)^2 + \left(y_u^p - y_j\right) \cdot \left(y_u^p + y_j - 2y\right)}{2 \cdot \left(x_u^p - x_j\right)} \\ y_o &= \frac{\left(y_u^p\right)^2 - \left(y_j\right)^2 + \left(x_u^p - x_j\right) \cdot \left(x_u^p + x_j - 2x\right)}{2 \cdot \left(y_u^p - y_j\right)} \end{aligned} \tag{13}$$

After calculating the values of r_{ij} and O_{ij}, the size of *safe region* $\mathfrak{F} s_i, p, s_j, p_u'$ can further be obtained. As shown in Fig. 8, the value of angle $\angle s_i O_{ij} s_j$ can be derived with two results according to the value of angle $\angle s_i p_u' s_j$. That is,

$$\angle s_i O_{ij} s_j = \begin{cases} 360^\circ - 2 \cdot \angle s_i p_u' s_j, & \textit{if } \angle s_i p_u' s_j > 90^\circ \\ 2 \cdot \angle s_i p_u' s_j, & \textit{otherwise.} \end{cases} \tag{14}$$

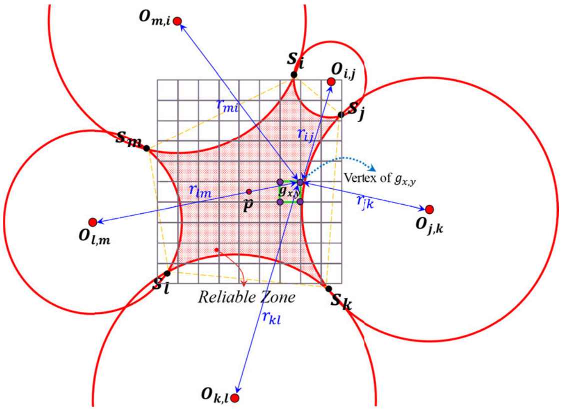


FIGURE 9. Example of ***Reliable zone.***

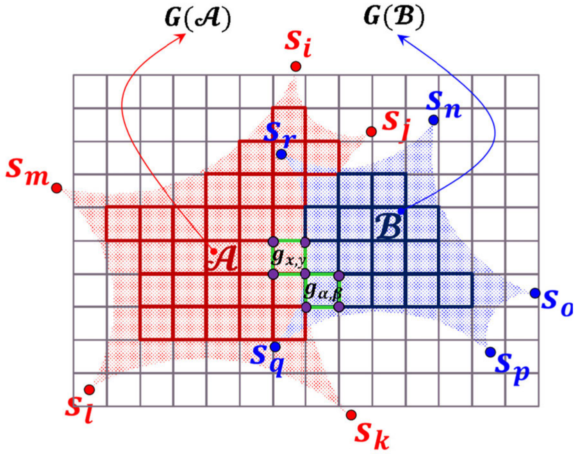


FIGURE 10. Example of ***cover group G*** **(Φ).**

According to Equs. (10) and (14), the size of *safe region* $\mathcal{F}s_i, p, s_j, p'_u$ can be evaluated by applying Equ. (15) if $\angle s_i p'_u s_j > 90$. Otherwise, the size of *safe region* $\mathcal{F}s_i, p, s_j, p'_u$ can be evaluated by applying Equ. (16).

$$\pi\left(r_{ij}\right)^2 \frac{360^\circ - 2\cdot\angle s_i p'_u s_j}{360} - \frac{1}{2}\left(r_{ij}\right)^2 \sin\left(360^\circ - 2\cdot\angle s_i p'_u s_j\right) \tag{15}$$

$$\pi\left(r_{ij}\right)^2 \frac{2\cdot\angle s_i p'_u s_j}{360} - \frac{1}{2}\left(r_{ij}\right)^2 \sin\left(2\cdot\angle s_i p'_u s_j\right) \tag{16}$$

Following the similar procedures from (8) to (16), the *reliable zone* can be obtained by the determined *safe region* of each pair of neighboring sensors, including pairs (s_i, s_j), (s_j, s_k), (s_k, s_l), (s_l, s_m), and (s_m, s_i), as shown in Fig. 9. The grid $g_{x,y}$ is guaranteed to be the full-view grid if it is inside the *reliable zone*. That is, as shown in Fig. 9, the distances between each of the vertices of $g_{x,y}$ and each of all circle centers, including the $O_{i,j}$, $O_{j,k}$, $O_{k,l}$, $O_{l,m}$, and $O_{m,i}$, should be greater or equal to every radius of circumscribed circles, including the r_{ij}, r_{jk}, r_{kl}, r_{lm}, and r_{mi}.

D. FULL-VIEW BARRIER CONSTRUCTION PHASE

Recall that, the *full-view* barrier can be constructed by the composition of a series of connected *full-view* grids from boundaries b_{left} to b_{right} in *WGM*. In this section, a *weighted graph* composed of vertex and edge sets will be constructed according to the connected relationship between *full-view* grids. Then, based on the constructed *weighted graph*, the goal (1) that aim to construct a maximum number of *full-view* barriers can be achieved while satisfying the constraints (2), (3), and (5). The following presents the formation of *weighted graph* and the construction of *full-view* barrier.

1) FORMATION OF WEIGHTED GRAPH

Let *Full-View Sensor Set of Grid* $g_{x,y}$, denoted by $S^{FV}_{x,y}$, represent a set of visual sensors that can support grid $g_{x,y}$ with *full-view* coverage. A set of grids is said to be *joinable* if they have the same *full-view sensor set*. A *Cover Group* $G(\Phi)$ of a *joinable grid set* Φ is a collection of IDs of those visual sensors that satisfy

$$G(\Phi) = \left\{ i \mid s_i \in S^{FV}_{x,y}, g_{x,y} \in \Phi \right\}. \tag{17}$$

Two cover groups $G(\mathcal{A})$ and $G(\mathcal{B})$ are said to be *neighboring* if there exist $g_{x,y} \in \mathcal{A}$ and $g_{\alpha,\beta} \in \mathcal{B}$ having at least one common point or side of grid. Let $|G(\Phi)|$ denote the number of visual sensors in $G(\Phi)$. Fig. 10 gives an example of $G(\Phi)$. As shown in Fig. 10, there are 10 visual sensors, including the s_i, s_j, s_k, s_l, s_m, s_n, s_o, s_p, s_q and s_r deployed in monitored region R. The grids that are *full-view* covered by visual sensors s_i, s_j, s_k, s_l, and s_m can be collected as *joinable grid set* $\mathcal{A}$, denoted by $G(\mathcal{A}) = \{i, j, k, l, m\}$. Besides, since there exists 5 visual sensors in $G(\mathcal{A})$, the $|G(\mathcal{A})|$ is 5. Similarly, we have $G(\mathcal{B}) = \{n, o, p, q, r\}$ and $|G(\mathcal{B})| = 5$. In addition, since $G(\mathcal{A})$ and $G(\mathcal{B})$ have at least one common point or side of grid such as $g_{x,y} \in \mathcal{A}$ and $g_{\alpha,\beta} \in \mathcal{B}$, they are *neighboring*.

As a result, we transform the cover groups to a *weighted graph* $G^{weight} = (V, E)$, where V and E denote the vertex set and edge set, respectively. Each vertex v_i in V represents the *cover group i*. There exists an edge $e_{i,j} = \left(v_i, v_j\right)$ in E between vertices v_i and v_j if *cover groups i* and *j* are *neighboring*. Let $w^{edge}_{i,j}$ denote the *weight value* of an edge $e_{i,j} = \left(v_i, v_j\right)$. The value of $w^{edge}_{i,j}$ is the number of relative complement of $G(i)$ with respect to $G(j)$. That is, $w^{edge}_{i,j} = |G(i) \setminus G(j)|$.

Fig. 11 gives an example of the *weighted graph*. As shown in Fig 11, the *weighted graph* composes of six vertices, including the v_1, v_2, v_3, v_4, v_5, and v_6. There are several edges composed of any two *neighboring* vertices such as vertices v_1 and v_2. Since $G(1) \setminus G(2) = \{f, g, h\}$, $w^{edge}_{1,2} = |G(1) \setminus G(2)| = 3$. Similarly, we have $w^{edge}_{2,3} = 4$, $w^{edge}_{1,4} = 5$, $w^{edge}_{4,5} = 2$, $w^{edge}_{5,6} = 3$, and so on.

Let $\left(x^v_i, y^v_i\right)$ denote the coordinates of v_i. The values of $\left(x^v_i, y^v_i\right)$ is represented by the coordinates of a grid that can be *full-view* covered by all visual sensors in $G(i)$ and located closest to the boundary W_{right}. Once there are more than one candidate *full-view* grid located closest to W_{right},

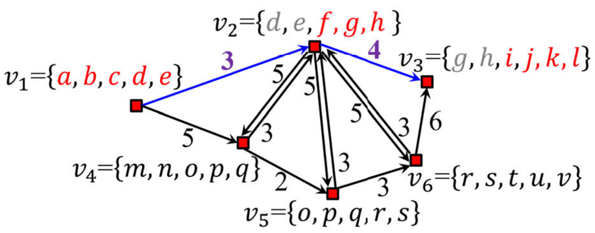


FIGURE 11. Construction of *weighted graph.*

the coordinates of the grid that is further located closest to the boundary L_{top} will become the coordinates of v_i. Let l denote the *edge length* of a grid in given region $R = L \times W$. Let v_s denote the *source vertex* which satisfies the condition that $\exists g_{x,y} \in G(s)$, where $x = 1$ and $1 \leq y \leq \left\lceil W/_l \right\rceil$. Similarly, let v_d be the *destination vertex* which satisfies the condition that $\exists g_{x,y} \in G(d)$, where $x = \left\lceil L/_l \right\rceil$ and $1 \leq y \leq \left\lceil W/_l \right\rceil$.

Let notation $\mathcal{P}_{s,d} = \left\{ e_{s,s+1}, e_{s+1,s+2}, \ldots, e_{d-1,d} \right\}$ be a path that composed of edges from $e_{s,s+1}$ to $e_{d-1,d}$ and discovered on G^{weight}. The path $\mathcal{P}_{s,d}$ discovered on G^{weight} represents that the corresponding coverage area satisfies *full-view* coverage. More specifically, the path $\mathcal{P}_{s,d}$ can also be treated as a full-view barrier. For example, consider Fig. 11. Assume that the path $\mathcal{P}_{s,d} = \left\{ e_{1,2}, e_{2,4}, e_{4,5}, e_{5,6}, e_{6,3} \right\}$. This path exactly represents the corresponding coverage area satisfies *full-view* coverage and hence also can be treated as a *full-view* barrier.

2) CONSTRUCTION OF FULL-VIEW BARRIER

To explore the path(*full-view* barrier) on given G^{weight}, in each vertex, a visual sensor that has the maximum number of neighbors is selected to be the *Decision Maker* (DM) which is responsible for executing the operations of the proposed FBCA. In case that there are more than one visual sensor satisfying the DM constraint, the visual sensor with largest ID will play the role of DM. Let notations v_i^{pre}, v_i^{curr}, and v_i^{next} denote the previous, current, and next considered vertex v_i by the proposed FBCA, respectively. To construct a path on given G^{weight}, the *source vertex* v_s is firstly treated as v_s^{curr}. A set, denoted by $N(v_i^{curr})$, is said to be the *Candidate Set* of v_i^{curr} is defined in the following.

$$N\left(v_i^{curr}\right) = \left\{ v_j \mid \exists \left(v_i^{curr}, v_j\right) \in E \right\} \tag{18}$$

The DM of v_s^{curr} will select an appropriate v_j from $N(v_s^{curr})$ to play the role of v_j^{next} if the value of $w_{s,j}^{edge}$ is minimal. This is because that selecting the v_j with minimal value of $w_{s,j}^{edge}$ indicates that the number of awaked visual sensors can be minimized during each selection of next vertex. Subsequently, an edge $e_{s,j} = (v_s^{curr}, v_j^{next})$ will be included in the path $\mathcal{P}_{s,d}$. That is, $\mathcal{P}_{s,d} = \left\{ e_{s,j} \right\}$. After accomplishing the first edge, the DM of v_s^{curr} will hand over the authority of next edge construction to the DM of v_j^{next}. The roles of v_s^{curr} and v_j^{next} will be changed to v_s^{pre} and v_j^{curr}, respectively. The DM of v_j^{curr} will execute the similar operations done by DM of v_s^{pre}. The edge construction will be executed repeatedly until a *destination vertex* has been included in *path* $\mathcal{P}_{s,d}$. Meanwhile, the path construction has been finished. Since the edge selection considers the edge with minimal *weight value*, the *path* is likely to be constructed by minimum number of visual sensors. This also implies that the number of *full-view* barriers constructed by the proposed mechanism is likely maximized.

The example given in Fig. 11 shows the construction of *full-view* barrier. In Fig. 11, the *source vertex* v_1 can be treated as v_1^{curr}. The DM of v_1^{curr} will select vertex v_2 from $N\left(v_1^{curr}\right) = \{v_1, v_4\}$ to play the role of v_2^{next} because the value of $w_{1,2}^{edge}$ is minimal as comparing with the $w_{1,4}^{edge}$. Meanwhile, an edge $e_{1,2} = (v_1^{curr}, v_2^{next})$ will be included in *path* $\mathcal{P}_{1,d} = \left\{ e_{1,2} \right\}$. The DM of v_1^{curr} will hand over the authority of next edge construction to the DM of v_2^{next}. The roles of v_1^{curr} and v_2^{next} will be changed to v_1^{pre} and v_2^{curr}, respectively. The DM of v_2^{curr} will further execute the similar operations done by DM of v_1^{pre}. As a result, we have the path $\mathcal{P}_{1,3} = \left\{ e_{1,2}, e_{2,3} \right\}$.This path $\mathcal{P}_{1,3}$ represents that the corresponding coverage area, as shown in Fig. 11. It also satisfies *full-view* coverage and hence can be treated as a *full-view* barrier.

V. SIMULATION

This section compares the performance of our proposed algorithm FBCA, and the existing algorithms Shortest Path (SP) and Minimum Camera Barrier Coverage Problem (MCSPS) in terms of the number of reduced grids, number of required sensors and coverage probability. The SP algorithm [15] proposed sub-regions which can be transformed into a graph. Based on the graph, their algorithm verifies whether each vertex satisfies the full-view coverage. Based on the weighted directed graph, an approach called MCSPS [16] can find a camera barrier which contains the minimum number of camera sensors. The following firstly shows the simulation model and then discusses the simulation results.

A. SIMULATION MODEL

The simulation parameters are given in Table 1. In the experimental study, the MATLAB is used as the simulation tool. The following illustrates the parameters considered in the simulation environment. The area size of the monitoring region is $100m \times 100m$. The number of camera sensors that are randomly deployed in the monitoring region varies from 500 to 3500. The field of view angle, denoted by θ, varies from $\pi/3$ to $2\pi/3$. The camera width varies from 6m to 12m. The edge length of each grid is set at 2m, 4m, 6m and 8m. The simulation parameters are given in Table 1. The following gives the setting of parameters in the experiments.

B. SIMULATION RESULTS

Recall that the first phase of proposed FBCA aims to remove the impossible grids for reducing the computational complexity. Fig. 12 shows the number of grids reduced by the

TABLE 1. Simulation parameters.

Parameter	Value
Simulator	Matlab R2018b
Node deployment	Random
Monitoring region	$100m \times 100m$
Number of Visual Sensors	$500 - 3500$
Grid size	$2m$,$4m$,$6m$,$8m$
Camera width	$6m$−$12m$
FoV Angle	$\pi/3 - 2\pi/3$

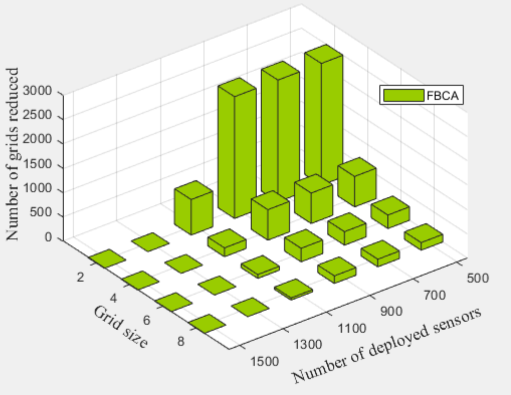


FIGURE 12. Performance of the proposed FBCA mechanism in terms of number of reduced grids.

proposed FBCA. The number of deployed sensors is varied ranging from 500 to 1500 and the grid size is varied ranging from $2m$ to $8m$. As shown in Fig. 12, the number of reduced grids decreases with the number of deployed sensors. This occurs because a grid can be covered by more sensors when the number of deployed sensors increases. Since each grid has a larger coverage, it has a higher opportunity to have full view. This implies that more grids are satisfied with the full view requirement. As a result, most grids will not be deleted by the proposed FBCA.

On the other hand, the number of reduced grids is decreased with the grid size. It is because that more visual sensors are required to coverage a grid with large grid size. When the number of the visual sensors is fixed, fewer grids can be full covered if the grid size grows. This also indicates that fewer grids can be reduced. As shown in Fig. 12, we can observe that the proposed FBCA reduces 729, 182, 80 and 45 grids, when the grid size is $2m$, $4m$, $6m$ and $8m$ at the 1100 number of deployed sensors respectively.

Fig. 13 compares the number of reduced grids with a different number of deployed sensors and different camera angles. The number of deployed sensors varied from 500 to 5000 whereas the camera angle varied from $\pi/3$ to $2\pi/3$. In general, the result of Fig. 13 is similar to that of Fig. 12. The number of reduced grids decreases with the number of deployed sensors.

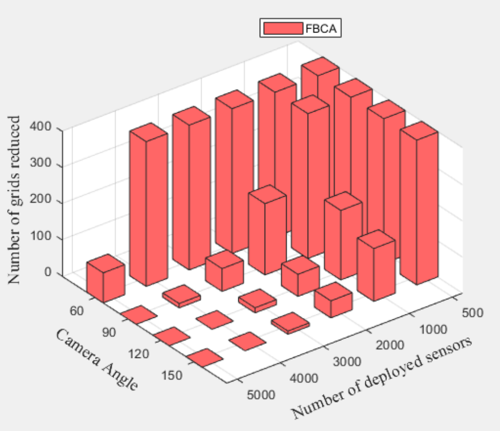


FIGURE 13. Performance of the proposed FBCA mechanism in terms of number of reduced grids.

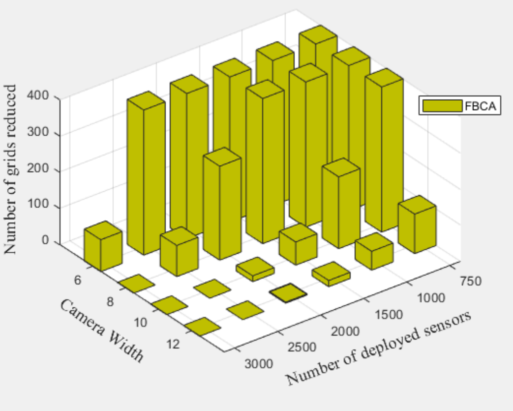


FIGURE 14. Performance of the proposed FBCA mechanism in terms of number of reduced grids.

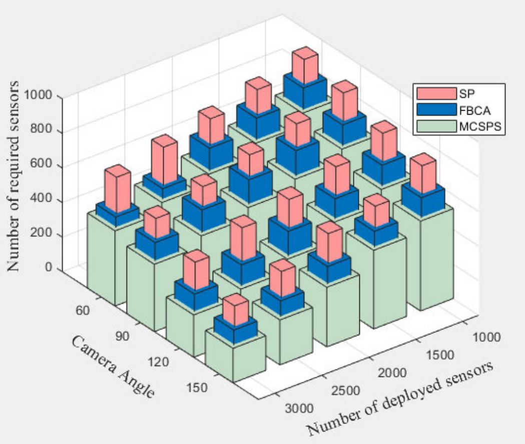


FIGURE 15. Comparison of the SP, proposed FBCA and MCSPS in terms of the number of required sensors.

On the other hand, the number of reduced grids decreases with the camera angle. When the number of sensors is fixed and the camera angle increases, each sensor can have a larger coverage area. Therefore, each grid is much easier to be covered by the minimum required number of sensors. Hence most of grids will not be deleted by the proposed FBCA. More specifically, the proposed FBCA reduces 83, 197, 192 and

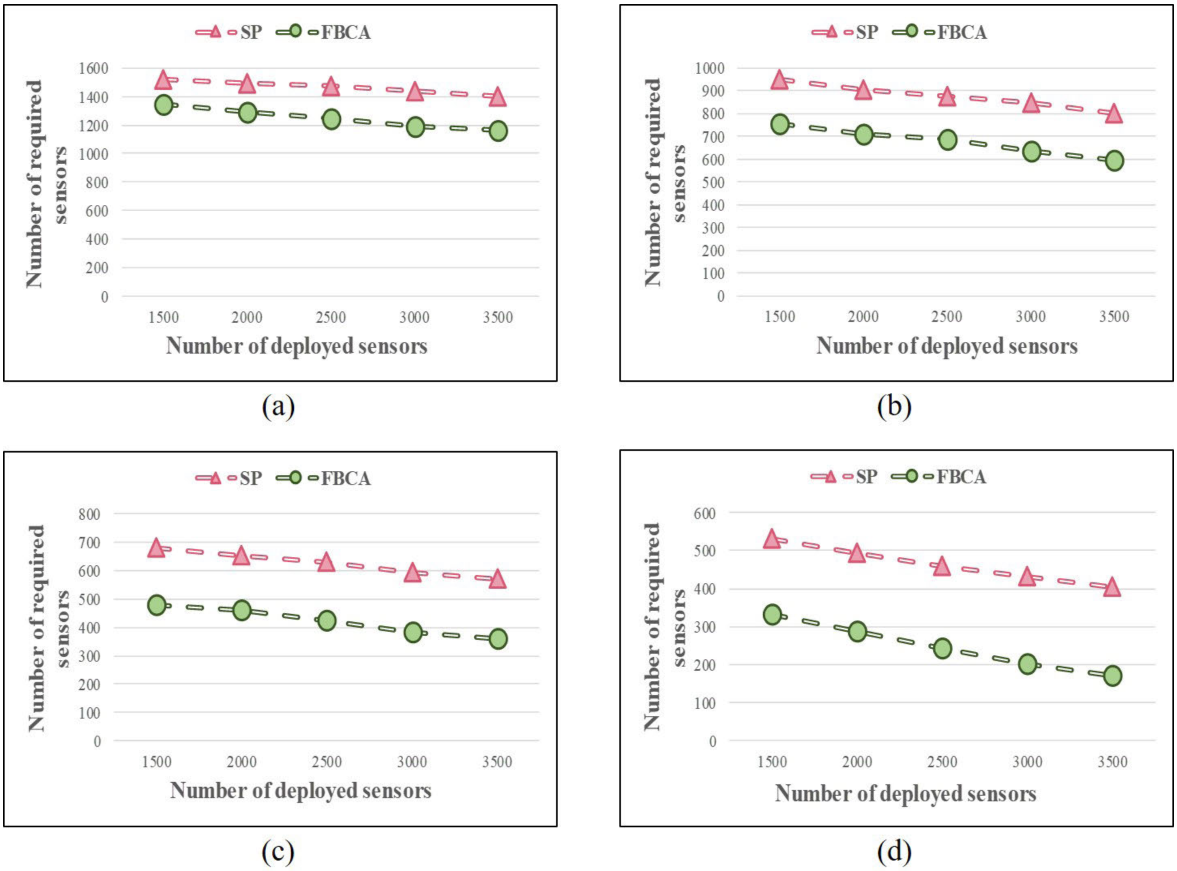


FIGURE 16. Performance comparison of *SP* and *FBCA* in terms of number of required sensors by varying the camera width. (a) Camera width r = 6*m*. (b) Camera width r = 8*m*. (c) Camera width r = 10*m*. (d) Camera width r = 12*m*.

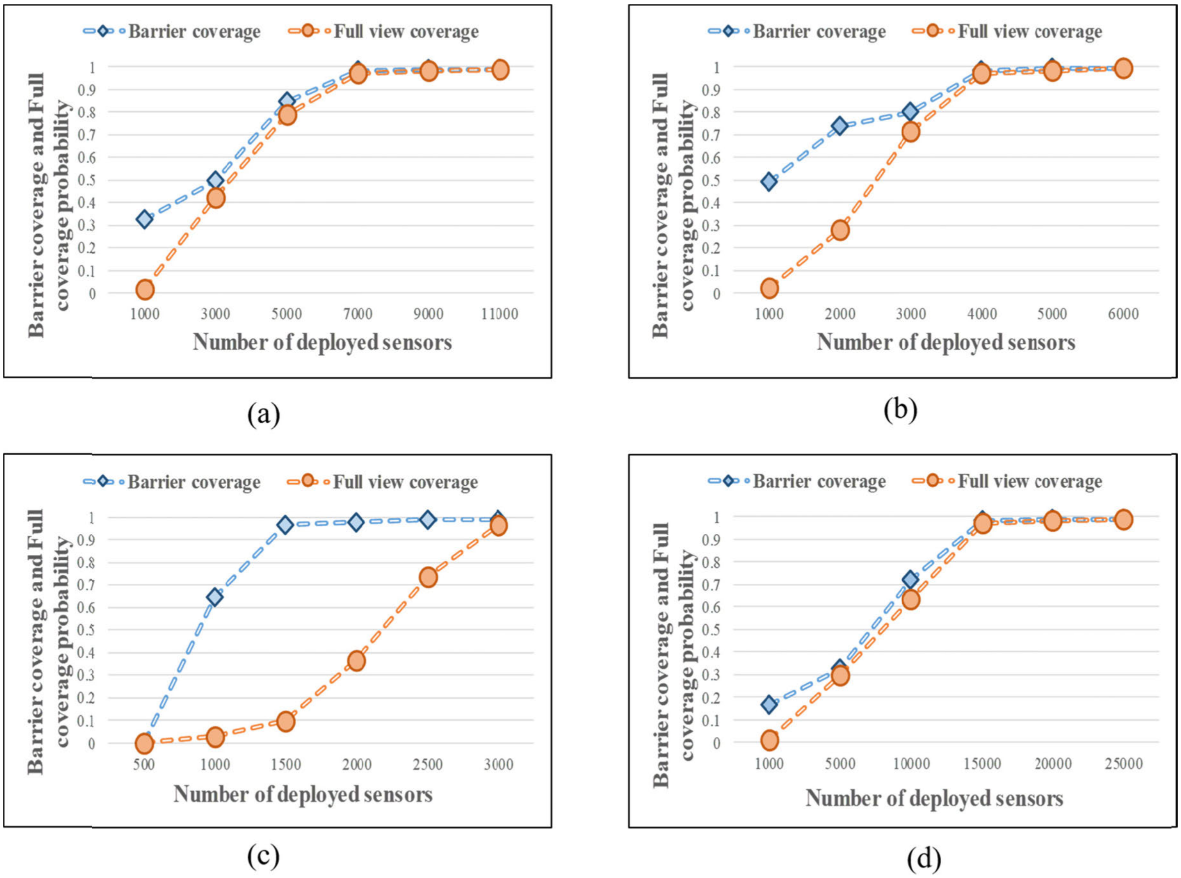
146 grids, when the number of deployed sensors are 5000, 2000, 1000 and 1000 at the angles 60, 90, 120 and 150 respectively.

Fig. 14 shows the number of reduced grids by considering two parameters: the number of deployed sensors and the camera width. The number of deployed sensors is varied from 750 to 3000 whereas the camera width varied from 6*m* to 12*m*. In general, the number of reduced grids decreases with the number of deployed sensors.

On the other hand, the number of reduced grids decreases with the camera width. When the camera width of each sensor increases, each sensor can have a larger coverage area. Therefore, each grid can easily to be covered by the required number of sensors and satisfy the required criteria. As a result, proposed FBCA can reduce fewer grids. As shown in Fig. 14, the proposed FBCA reduces 88, 259, 198 and 109 grids when the camera width is fixed at 6*m*, 8*m*, 10*m* and 12*m*, respectively.

The second phase of the proposed FBCA aims to find the set of sensors for constructing the *full view* barrier. Fig. 15 compares the MCSPS, SP and proposed FBCA in terms of the number of required sensors. It depicts that the number of required sensors for constructing a *full-view* barrier by varying the camera angle ranging from $\pi/3$ to $2\pi/3$ in a fixed region $R = 100m$ X $100m$. The number of deployed sensors varied from 1000 to 3000. As shown in Fig. 15, the MCSPS algorithm requires the least number of sensors, as compared with the FBCA and SP. The main reason is that the MCSPS algorithm transforms the monitoring region into the weighted graph. Based on the graph, all the possible *full-view* barriers are explored by the proposed algorithm. However, MCSPS has high computational and communication costs since it is centralized. Compared with both the MCSPS and SP, the performance of the proposed FBCA is similar to the optimal solution MCSPS and outperforms the SP, in terms of a number of the required sensors. In addition, both the computation and communication costs of each visual sensor are significantly reduced by applying the proposed decentralized FBCA.

Given a fixed number of sensors ranging from 1500 to 3500, Fig. 16 investigates the number of required sensors for the SP algorithm and proposed FBCA by varying the camera width. In Figs. 16(a), 16(b), 16(c) and 16(d), camera widths are set by 6*m*, 8*m*, 10*m* and 12*m*, respectively. The FBCA outperforms SP in terms of the number of sensors in all cases. This occurs because when the proposed FBCA algorithm constructs a path, it will wake up fewer additional visual sensors. Therefore, FBCA requires a smaller number



(a)

(b)

(c)

(d)

FIGURE 17. Performance comparison of *Barrier coverage* and *Full coverage* by varying the camera angle. (a) Camera angle $\theta = 2\pi/3$. (b) Camera angle $\theta = \pi/2$. (c) Camera angle $\theta = \pi/3$. (d) Camera angle $\theta = \pi/6$.

of required sensors, as compared to SP. In general, the number of required sensors decreases with the camera width. The main reason is that if the camera width of each visual sensor increases, each sensor can have a larger coverage area. Therefore, it requires a smaller number of sensors to achieve the full-view requirement. More specifically, when the number of deployed sensors is 2000 and the camera widths are varied by $6m$, $8m$, $10m$ and $12m$, the proposed FBCA requires 1290, 712, 459 and 288 sensors, respectively. The third phase of the proposed FBCA aims to construct a barrier for full view. However, when the number of cameras are small, the full view coverage purpose cannot be achieved. The constructed barrier can only support barrier coverage which can detect the intruder but unable to have full view of the intruder. Fig. 17 compares the probability of barrier coverage and full view coverage by varying the camera angle. The target field R is a $20m$ X $100m$ rectangle region. In Fig. 17(a), the camera angle is set by $2\pi/3$. The numbers of deployed sensors are varied from 500 to 3000. In Figs. 17(b), 17(c) and 17(d), camera angles are reduced to $\pi/2$, $\pi/3$ and $\pi/6$, respectively. Since the camera angle is reduced, the number of camera sensors should be increased for guaranteeing that the full view coverage is also possible. Therefore, the number of deployed sensors in Figs. 17(b), 17(c) and 17(d) are set by 1000 to 6000, 1000 to 11000 and 1000 to 25000. In general, the experimental results of four Figures depict that the coverage probability for the barrier coverage is much higher than that of full view coverage. As shown in Fig. 17(a), when the number of deployed sensors are 1500, the barrier coverage probability is almost 1. On the contrary, the number of sensors required for full view coverage is at least 3000. The number of sensors required for supporting full view is increased when the camera angle is decreased. Therefore, as shown in Figs. 17(b), 17(c) and 17(d), there are 4000, 6000 and 12000 sensors required for achieving full view coverage, respectively.

VI. CONCLUSION

In this paper, the FBCA mechanism is proposed aiming to construct as many as possible full-view barriers. The constructed full-view barriers can work in turns and each of them can capture the complete face of the intruder in any facing direction. The proposed FBCA is a decentralized solution which exploits scalability and flexibility for supporting the changes of network size and the network topology. Experimental studies show that the proposed FBCA achieves better performance than existing SP, and likely approaches to the optimal solution (MCSPS) in terms of the number of required sensors for constructing a barrier with full view coverage.

The proposed FBCA mechanism has an extensive range of applications. In the border surveillance, our algorithm can be used to detect the intruder crossing the boundary. In security monitoring applications such as military and hospitals etc., it can monitor the people entering into the dangerous regions. There are two limitations with our proposed mechanism for the real applications. The first one is that sensors should be homogenous. The applications where heterogeneous sensors are needed will constrain our mechanism in monitoring a complex environment. The second one is the fixed sensing range of each sensor. If the sensor is deployed far away from the boundary curve, with the fixed sensing range, there is no contribution to the monitoring applications. The future research issue is considering the heterogeneous sensors with adjustable sensing range.

REFERENCES

[1] D. Thomas, R. Shankaran, M. Orgun, M. Hitchens, and W. Ni, "Energy-efficient military surveillance: Coverage meets connectivity," *IEEE Sensors J.*, vol. 19, no. 10, pp. 3902–3911, May 2019.

[2] Y. Wu, Y. Wang, and G. Cao, "Photo crowdsourcing for area coverage in resource constrained environments," in *Proc. IEEE INFOCOM*, Atlanta, GA, USA, May 2017, pp. 1–9.

[3] D. Tao and T.-Y. Wu, "A survey on barrier coverage problem in directional sensor networks," *IEEE Sensor J.*, vol. 15, no. 2, pp. 876–885, Feb. 2015.

[4] M. Karatas, "Optimal deployment of heterogeneous sensor networks for a hybrid point and barrier coverage application," *Comput. Netw.*, vol. 132, pp. 129–144, Feb. 2018.

[5] M. Khanjary, M. Sabaei, and M. R. Meybodi, "Barrier coverage in adjustable-orientation directional sensor networks: A learning automata approach," *Comput. Elect. Eng.*, vol. 72, pp. 859–876, Nov. 2018.

[6] S. Kumar, T. H. Lai, and A. Arora, "Barrier coverage with wireless sensors," in *Proc. ACM MobiCom*, Aug. 2005, pp. 284–298.

[7] S. Kumar, T. H. Lai, M. E. Posner, and P. Sinha, "Maximizing the lifetime of a barrier of wireless sensors," *IEEE Trans. Mobile Comput.*, vol. 9, no. 8, pp. 1161–1172, Aug. 2010.

[8] A. Chen, S. Kumar, and T. H. Lai, "Local barrier coverage in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 9, no. 4, pp. 491–504, Apr. 2010.

[9] C. F. Huang and Y. C. Tseng, "The coverage problem in a wireless sensor network," in *Proc. ACM WSNA*, 2003, pp. 115–121.

[10] G. Yang and D. Qiao, "Multi-round sensor deployment for guaranteed barrier coverage," in *Proc. IEEE INFOCOM*, Mar. 2010, pp. 1–9.

[11] R. Ghazalian, A. Aghagolzadeh, and S. M. H. Andargoli, "Wireless visual sensor networks energy optimization with maintaining image quality," *IEEE Sensors J.*, vol. 17, no. 13, pp. 4056–4066, Jul. 2017.

[12] X. Yang, Y. Wen, D. Yuan, M. Zhang, H. Zhao, and Y. Meng, "3-D compression-oriented image content correlation model for wireless visual sensor networks," *IEEE Sensors J.*, vol. 18, no. 15, pp. 6461–6471, Aug. 2018.

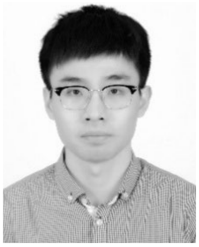
[13] Z. Wang, J. Liao, Q. Cao, H. Qi, and Z. Wang, "Achieving k-barrier coverage in hybrid directional sensor networks," *IEEE Trans. Mobile Comput.*, vol. 13, no. 7, pp. 1443–1455, Jul. 2014.

[14] Y. Pei and F. Hou, "Social-aware incentive mechanism for full-view covered video collection in crowdsensing," *IET Commun.*, vol. 12, no. 20, pp. 2600–2608, Dec. 2018.

[15] Y. Wang and G. Cao, "Barrier coverage in camera sensor networks," in *Proc. ACM MobiHoc*, May 2011, Art. no. 12.

[16] H. Ma, M. Yang, D. Li, Y. Hong, and W. Chen, "Minimum camera barrier coverage in wireless camera sensor networks," in *Proc. IEEE INFOCOM*, Mar. 2012, pp. 217–225.

[17] Y. Zhang, S. He, and J. Chen, "Data gathering optimization by dynamic sensing and routing in rechargeable sensor networks," *IEEE/ACM Trans. Netw.*, vol. 24, no. 3, pp. 1632–1646, Jun. 2015.



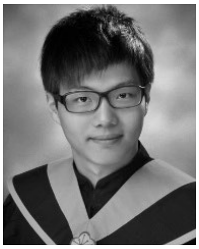
PEI XU received the M.S. degree in control theory and control engineering from Anhui Polytechnic University, Wuhu, Anhui, China, in 2014. He is currently pursuing the Ph.D. degree with the School of Computer Science and Technology, Anhui University, Hefei, Anhui. His current research interests include intelligent interaction and artificial intelligence.

I-HSIUNG CHANG received the Ph.D. degree in education from National Chengchi University, Taiwan, in 2012. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Information Engineering, Tamkang University, Taiwan. His current research interests include the joint of education and information technologies in the AI-IoT, robot, and big data.

CHIH-YUNG CHANG (M'05) received the Ph.D. degree in computer science and information engineering from National Central University, Taiwan, in 1995.

He is currently a Full Professor with the Department of Computer Science and Information Engineering, Tamkang University, New Taipei, Taiwan. His current research interests include the Internet of Things, wireless sensor networks, artificial intelligence, and deep learning. He has served as an Associate Guest Editor for several SCI-indexed journals, including the *Journal of Internet Technology* (JIT), from 2004 to 2008, the *Journal of Information Science and Engineering* (JISE), in 2008, *Telecommunication Systems* (TS), in 2010, *IET Communications*, in 2011, the *International Journal of Ad Hoc and Ubiquitous Computing* (IJAHUC), from 2011 to 2014, and the *International Journal of Distributed Sensor Networks* (IJDSN), from 2012 to 2014.

BHARGAVI DANDE received the M.S. degree in computer science and information engineering from Tamkang University, Taipei, Taiwan, in 2019, where she is currently pursuing the Ph.D. degree in computer science and information engineering. Her current research interests include the Internet of Things, wireless sensor networks, and artificial intelligence.



CHIH-YAO HSIAO received the B.S. degree from I-Shou University, Taiwan, in 2009, and the M.S. degree from Tamkang University, Taiwan, in 2011, all in computer science and information engineering, where he is currently pursuing the Ph.D. degree in computer science and information engineering. He received several scholarship grants in Taiwan and has participated in many projects related to wireless sensor networks, vehicular ad hoc networks, the Internet of Things, and software-defined networks. His research interests include the Internet of Things, wireless sensor networks, ad hoc wireless networks, and vehicular ad hoc networks.