ARTICLE IN PRESS
Ain Shams Engineering Journal (2016) **xxx**, xxx–xxx

Ain Shams University
Ain Shams Engineering Journal
www.elsevier.com/locate/asej
www.sciencedirect.com

ELECTRICAL ENGINEERING

A novel approach using optimum camera actuation in event boundary detection method for redundant data minimization

Sushree Bibhuprada B. Priyadarshini *, **Suvasini Panigrahi**

Veer Surendra Sai University of Technology (VSSUT), Burla, Sambalpur, Odisha 768018, India

Received 16 July 2015; revised 13 March 2016; accepted 14 April 2016

KEYWORDS

Field of view;
Depth of field;
Boundary sensors;
Binary event measurement parameter;
Redundancy ratio

Abstract The necessity of having optimum camera activation is rapidly increasing in distributed environment as well as wireless sensor networks. In the current research, we have studied the event boundary detection approach for redundant data minimization by actuating less number of cameras. The study reveals that some of the cameras those are present outside the exact event boundary are activated unnecessarily being informed from the boundary scalars regarding the event occurrence. This unnecessary activation of outer cameras leads to additional energy expenditure and redundant data transmission. In this paper, we have proposed a novel approach to maintain such unnecessarily actuated cameras in turned off state. The experimental evaluation in terms of less camera activation, minimized redundancy ratio, enhanced coverage ratio and reduced energy consumption, obtained from the investigation justifies the effectiveness of the proposed approach as compared to another approach recently proposed in the literature.

1. Introduction

In this modern era of sensor technology, wireless multimedia sensor networks (*WMSNs*) have drawn significant amount of attention and interest for the researchers as well as scientists. The reason behind such motivation towards *WMSNs* is that they use video sensors (i.e. camera sensors) along with scalar sensors. Therefore, multimedia data such as video and audio streams can easily be retrieved and necessary processing can be done upon the data collected. However, battery operated cameras are used in *WMSNs* to capture any kind of event occurring in the interested monitored region. Normally, scalars remain in active state always and hence they can sense the monitored area all the time whereas the cameras are kept in turned off condition. Further, the cameras are activated only when they are informed regarding occurrence of any kind of event by their corresponding scalars. The camera sensors have two fundamental parameters – *field of view* (*FOV*) and *depth of field* (*DOF*). *FOV* refers to the angle at which a camera can capture the accurate image of any object whereas *DOF*

* Corresponding author. Mobile: +91 8908465489.
E-mail addresses: bimalabibhuprada@gmail.com (S.B.B. Priyadarshini), suvasini26@gmail.com (S. Panigrahi).

Peer review under responsibility of Ain Shams University.

ELSEVIER Production and hosting by Elsevier

http://dx.doi.org/10.1016/j.asej.2016.04.013
2090-4479 © 2016 Faculty of Engineering, Ain Shams University. Production and hosting by Elsevier B.V.

represents the distance within which a camera can trap the accurate image of an object [1].

A study on distributed camera activation proposed by Newell and Akkaya [1] reveals that when any sort of event occurs in a monitored region, it is at first detected by the scalars. Subsequently, the concerned scalars communicate the information regarding the occurring event to their corresponding cameras. Afterwards, the cameras collaboratively decide who among them are to be actuated (i.e. turned on). The corresponding cameras represent the cameras within whose $FOVs$ the scalar lies. The same paper [1] also discusses the event boundary detection method, which concentrates on detection of the occurring event by the scalars sitting at the boundary of occurring event. In this method, each sensor exchanges its binary event measurement parameter with its concerned neighbours. The binary parameter contains a value of either 1 or 0 based on whether it is a boundary node or not. As opposed to all the scalars, only the scalars which sit at the boundary (i.e., boundary scalars) of an event can inform the cameras regarding the occurrence of event. However, the problem is that some of the cameras which are present outside the exact event boundary are informed by the boundary scalars since their *DOFs* cover the concerned scalars. As a result, the cameras present outside the event boundary are unnecessarily activated. In the current research, a novel method is devised that leads to turn off of the unnecessarily activated cameras. This leads to reduced camera activation along with minimized redundancy ratio while affording enhanced event coverage. Basically, the used scenario is aimed for tracking the activities of living animals and plants (habitat monitoring) by using omni-directional cameras that are deployed to ensnare the occurring event information uniformly along all the directions.

This paper is organized as follows: Section 2 summarizes the related work done in the field of redundant data transmission and coverage of events. Section 3 includes the methodology used in proposed approach as well as the working steps of the proposed algorithm. Section 4 elaborates the simulation details along with results and discussions. Finally, in Section 5, we conclude the paper with directions for future research.

2. Related work

Several works have been carried out time and again in the field of redundant data minimization for optimum camera activation. Newell and Akkaya [1] suggested distributed approaches for camera activation, where the cameras collaboratively decide which among them are to be activated. Basically, two approaches are discussed in the paper – one method uses scalar count approach and another uses event boundary detection approach. In scalar count approach, the camera sensor having maximum number of event detecting scalars is activated first and the rest of the camera sensors are activated based on descending values of their scalar counts [1]. To get more accurate information regarding the occurring event, event boundary detection method was proposed. The theme of this method is that only the scalars those are present at the event boundary detect the event and they communicate their reading to their corresponding camera sensors.

The idea of cover-set [2] helps in monitoring all the targets in a monitored area. A novel and efficient algorithm is proposed for giving a sub-optimal solution to the problem. The concept of directional coverage approach given by Wang et al. [3] concentrates on individual targets associated with differentiated priorities, where the target information is to be captured. Further, another approach on path coverage is suggested in [4] in which the network coverage of two-dimensional area is analysed for random deployment of sensors. The work done in [5] gives emphasis on multiple directional cover set problem. Moreover, Girault [6] suggests an algorithm that proceeds in two passes for redundant data elimination. The work done in [7] proposes an algorithm based on data similarity, used for redundant data elimination. Toumpis and Tassiulas [8] elaborated an optimal strategy for sensor deployment. A redundant positioning architecture devised by Han et al. [9] discusses a novel architecture for processing vast amount of data from pervasive devises. An algorithm given in [10] provides required k coverage, where each point in a deployment field is covered by at least k sensors.

Although several works have been carried out for optimum camera activation and redundant data minimization, still the activation of only required number of cameras with adequate event coverage is a challenging problem. None of the works as discussed above have considered the unnecessary cameras activated due to the event boundary detection approach. In this paper, we have proposed a novel algorithm for event boundary detection method so as to turn on only the essential number of camera sensors while keeping the unnecessarily activated cameras in off state for achieving less energy expenditure and minimized amount of redundant data transmission.

3. Proposed approach

The event boundary detection method discussed in [1], proposes a scheme for camera actuation in which only the boundary scalars communicate accurate event information to their corresponding camera sensors. However, undesired camera activation takes place due to sensing of events by the scalars lying within the *DOFs* of outer cameras (i.e., cameras present in between the *event radius* (R) to ($R + DOF$)), which causes unnecessary activation of outer cameras. Due to such undesired camera activation, unnecessary energy consumption as well as redundant data transmission occurs. A scenario of event occurrence is demonstrated in Fig. 1, where multiple cameras are unnecessarily activated due to event information communication by the boundary scalars. The large circle with radius R around the event point represents the occurring event region. Initially, a binary event measurement parameter is exchanged among the sensors. Suppose, we consider the cameras $C1$ to $C10$ represented as squares in Fig. 1 that are present outside the exact event boundary. However, they are still activated since their *DOFs* cover some of the boundary scalars. The problem is that due to the activation of these outer cameras, more portion of non-event area and only a little fragment of the actual event region are covered, which could also be covered by some other inner cameras.

3.1. Methodology used in the proposed approach

In the proposed approach, we have improved the event boundary detection method [1] for minimizing the undesired camera activation taking place due to the cameras lying outside the event boundary (i.e. beyond event radius (R)). This is

Please cite this article in press as: Priyadarshini SBB, Panigrahi S, A novel approach using optimum camera actuation in event boundary detection method for redundant data minimization, Ain Shams Eng J (2016), http://dx.doi.org/10.1016/j.asej.2016.04.013

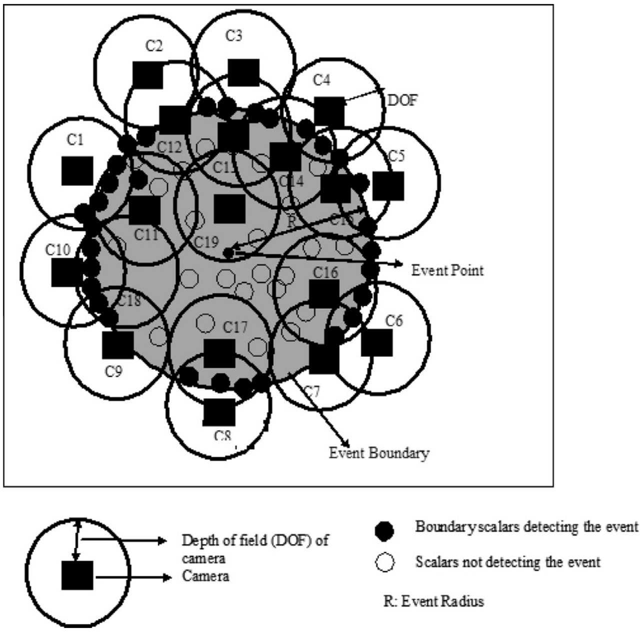

Figure 1 Event Detection in Event Boundary Approach.

accomplished by activating only the cameras present within the event boundary (R). In addition, our method is able to achieve adequate coverage of the event region which is expressed in terms of *Coverage Ratio* [1]. *Coverage Ratio* is defined as the ratio of total portions of event area covered by activated cameras with respect to the total area of occurring event, which is expressed as follows:

$$Coverage\ Ratio = \frac{tpe}{te} \tag{1}$$

where
tpe: total portions of event area covered by activated cameras.
te: total area of occurring event.

It is obvious that less is the number of cameras activated, less will be the amount of overlapping of $FOVs$ among cameras. This will result in less amount of redundant data transmission. In the current work, we express the redundant data in terms of *Redundancy Ratio*. *Redundancy Ratio* is defined as the ratio of total portions of overlapping areas of $FOVs$ of activated cameras covering the occurring event region to the total unique portions of event area that is covered by the activated cameras, which is expressed as follows:

$$Redundancy\ Ratio = \frac{tof}{tue} \tag{2}$$

where
tof: total portions of overlapping area of $FOVs$ of activated cameras.
tue: total unique portions of event area covered by activated cameras.

While considering the case of event tracking by cameras, the value of *Coverage Ratio* should be maximized and the *Redundancy Ratio* value should be minimized.

3.2. Proposed algorithm

Following are the working steps involved in the proposed approach:

- All the nodes (i.e. sensors) are deployed randomly in an area of interest. The cameras broadcast $MYCIM$ (*My Camera Information Message*) and scalars broadcast $MYSIM$ (*My Scalar Information Message*). Both these messages contain the *ids* and location information of the concerned sensors. Sufficiently a large number of nodes are assumed to be deployed such that each and every point of monitored region is covered.
- $MYFOV$ (*My Field of View Table*) is initialized to zero, which contains the number of scalars present within FOV of a camera sensor. The status of all the cameras is initialized to $FALSE$ which signifies that the concerned camera is in off state.
- Whenever an event takes place, the *boundary scalars* detect the event. Subsequently, they communicate their reading to their respective cameras by broadcasting a message $DETECTT$. All the nodes broadcast a *binary event measurement parameter*, which can either be 0 or be 1 for scalar sensors and 00 or 11 for camera sensors.
  Case 1: If the sensor is a *boundary scalar*, then it broadcasts parameter value 1; otherwise, it broadcasts value 0.
  Case 2: If the sensor is a *boundary camera* or if it is present within the exact event region then it broadcasts parameter value 11; otherwise, it broadcasts the parameter value as 00.
- Suppose, R be the event radius. Among the cameras containing binary parameter as 11, the *ids* of cameras present within distance (R) to ($R - DOF$) are kept in a $MAINTAIN$ table and the status of cameras present outside R is kept as $FALSE$ while deciding the camera activation for boundary nodes. The cameras maintained in $DETECTT$ table, which are informed by the boundary sensors (i.e. cameras sitting within distance ($R - DOF$) to (R)), broadcast $MYPOLYGON$ message. This message is sent from a camera sensor to one or more cameras that contain *ids* of scalars lying at the event boundary. Afterwards, the cameras match the scalar ids present in $MYFOV$ table with the ids present in $MYPOLYGON$ message. The cameras are then activated based on the common region shared between their $FOVs$ and the Event boundary region. The matching is done based on matching the scalar ids present within concerned camera's FOV and the scalar ids present within event boundary.
- At first, the camera having largest intersection area (i.e., the camera having maximum scalar ids common) is activated. If there is a tie between any two cameras, then any one of them can be activated first. The activated camera broadcasts $MYUPDATE$ message that contains the intersection region information. Subsequently, the rest of the cameras are activated based on their descending order of the common event area covered and according to the contents of $MYUPDATE$ message.

In this work, we have kept the cameras present outside event radius in turned off condition, which were activated unnecessarily in the previous work [1] as referred earlier. Therefore, we considered only the cameras present at distance

Please cite this article in press as: Priyadarshini SBB, Panigrahi S, A novel approach using optimum camera actuation in event boundary detection method for redundant data minimization, Ain Shams Eng J (2016), http://dx.doi.org/10.1016/j.asej.2016.04.013

$(R - DOF)$ to R as boundary cameras instead of $(R - DOF)$ to $(R + DOF)$, which are used to communicate event boundary information. Further, the cameras present within the exact event boundary (R) collaborate among themselves to take decision for activation. Moreover, the energy consumption and the amount of redundant data transmission are minimized in the proposed approach as experimentally demonstrated in the next section.

4. Performance evaluation

The performance of our proposed method has been assessed by carrying out experimental analysis and comparative exploration with another recently proposed approach [1] in the literature.

4.1. Simulation environment

The implementation of the proposed approach has been carried out in *C++* in *UBUNTU* platform using our own simulated scenario while keeping the perspective of real world sketch in mind. The sensors are modelled in the simulated scenario by considering the following assumptions: (i) All the cameras and scalars are randomly deployed and the sensors are assumed to have fixed positions as used in [1]. The random deployment of sensors is because of the fact that in real life scenarios the cameras are sprinkled randomly from air planes across the deployment region. Further, if we prefer to deploy the cameras manually by human being for avoiding the overlapping among field of views of cameras, it is not practically feasible on the part of human being to deploy nodes in far away remote inaccessible region, and (ii) the entire network consists of two types of sensors namely scalars and cameras. The number of scalars is taken to be much lesser than the number of cameras, (iii) the cameras are assumed to be omni-directional cameras that are capable of capturing images of objects uniformly along all the directions. Further, the use of omni-directional camera provides panoramic view of the occurring event; (iv) further, the nodes are battery operated nodes and those are both time synchronous and are equipped with processors to do complex processing operations and (v) the event region is assumed to be circular as done in case of the initial event boundary detection approach [1].

4.1.1. Data generation

The data generation procedure that has been adopted by us uses the following notations:

Notations	Meaning
noss:	total number of scalar sensors
nocs:	total number of camera sensors
sxco [*i*]:	array containing *x* coordinates of scalar sensors to be deployed
syco [*i*]:	array containing *y* coordinates of scalar sensors to be deployed
cxco [*j*]:	array containing *x* coordinates of camera sensors to be deployed
cyco [*j*]:	array containing *y* coordinate of camera sensors to be deployed
n and *m*:	Number of scalar and camera nodes to be deployed

The pseudocode for the data generation procedure has been depicted as follows:

```
//x and y coordinate position generation for scalars
  for(i = 0; i < noss; i++)
    {
      sxco[i] = rand()% (n + 1); //sxco[i] is in the range of 0 to n
      syco[i] = rand()%(n + 1); //syco[i] is in the range of 0 to n
      cout << sxco[i] << " " << syco[i ]< < endl;
    }
//x and y coordinate position generation for cameras
  for(j=0; j < nocs; j++)
    {
      cxco[j] = rand()%(m + 1);//cxco[j] is in the range of 0 to m
      cyco[j] =r and()%(m + 1);// cyco[j] is in the range of 0 to m
      cout << cxco[j] << " " << cyco[j] << endl;
    }
  for(i=0, j=0; i < noss, j < nocs; i++, j++)
    {
      if(cxco[j] = = sxco[i] && cyco[j] = = syco[i])
//if coordinate position of camera sensor and scalar sensor matches
      {
        cxco[j] = cxco[j] + 1;
//increment x-coordinate position of camera sensor by 1 to avoid overlapping of coordinate positions
        cout << cxco[j] << " " << cyco[j] << endl;
        cout << sxco[i] << " " << syco[i] << endl;
      }
  }
```

Several global variables are used for broadcasting messages time to time to enable information exchange between nodes through parameter passing while applying various user-defined functions. Based on the broadcasted messages exchanged among the camera nodes, all the nodes determine their neighbour's location according to the (x, y) coordinate value generated through *rand*(). The distance among the nodes is then estimated by applying *Euclidean distance measure* expressed as follows:

$$E.Dist(S_i, S_j) = \sqrt{(x_j - x_i)^2 + (y_j - y_i)^2} \quad (3)$$

where $E.Dist(S_i, S_j)$: *Euclidean distance* between sensor $S_i(x_i, y_i)$ and sensor $S_j(x_j, y_j)$.

Our simulator is developed in such a manner that it can handle the case of both fixed as well as mobile objects. In case of both kinds of objects, the entire event region is taken as circular considering the point of initial occurrence of event as the *event point* and its diameter is the maximum distance from this initial position to the distant point up to which event occurs. This maximum distant point can be determined based on the prevailing signal strength received by the sensors. The entire process of event location and radius determination for any kind of object is elaborated in the next subsection.

4.1.2. Determination of event location and the radius of event by the scalars

Whenever any kind of event takes place the scalar sensors detecting the event, receive signal of certain intensity. The signal strength goes on decreasing with increase in distance from the point of occurrence of event. When a scalar receives any signal, it broadcasts $MDATA$ message that contains the intensity of signal it received along with its time of receipt. Suppose, U_1: intensity of signal at the point of occurrence of event, R: event radius, V_1: intensity of signal at any point (other than the event point), where any scalar n is located, a_1: decelerating rate of signal intensity which is considered as constant since we are taking barrier free view of occurring event. Let p_1, q_1: represent any two scalars deployed, where p_1 is the scalar closer to the event than q_1 (which indicates intensity of signal received by p_1 is the more than that of q_1). Let u_1, v_1: intensities of signals received by p_1 and q_1 respectively. Suppose, Δt: time taken for the signal to travel from p_1 to q_1 which is expressed as follows:

$$\Delta t = t_2 - t_1 \tag{4}$$

where t_1 and t_2 are the time when the signals are received by p_1 and q_1 respectively. Since the intensity of signal received by q_1 is less than that of p_1 ($v_1 < u_1$) and a_1 is the decelerating rate of intensity (a is negative). Thus, we can express v_1 as follows:

$$v_1 = u_1 - a_1 \times \Delta t \tag{5}$$

$$\Rightarrow a_1 = (u_1 - v_1)/\Delta t \tag{6}$$

Subsequently, the scalar estimates the location of the other scalar and suppose this point be $m_1(X_i, Y_i)$ where signal intensity is the maximum from the $MDATA$ message i.e., this is the observable point where the event is considered to take place. This intensity is the value of U_1.

$$\text{Now, } V_1^2 = U_1^2 - 2 \times a_1 \times S_1, \text{ where } V_1 \leq U_1 \text{ (always)} \tag{7}$$

$$\text{or } \quad S_1 = (U_1^2 - V_1^2)/2a_1 \tag{8}$$

where S_1 is the distance of event point from any sensor n and signal intensity is V_1. The location of minimal signal strength receiving scalar is found out from the broadcasted message $MDATA$ received by several scalars and let this point be $g(X_k, Y_k)$. At the time of deployment, a threshold value for the intensity of signal received by the scalar is set to determine whether the intensity of the received signal is significant enough to take decision that the event information captured is relevant or not. From the $MDATA$ message, the coordinate position of the scalar having received signal intensity value equal to or immediately greater than the threshold value is found out. Suppose this point is denoted as $h_1(X_j, Y_j)$. The Euclidian distance from point $m_1(X_i, Y_i)$ to $h_1(X_j, Y_j)$ is the value of event radius R given by the following expression:

$$R = \sqrt{(X_j - X_i)^2 + (Y_j - Y_i)^2} \tag{9}$$

In this manner, the scalar determines the location and radius of occurring event. The distributed event boundary approach proposed in [1] is considered as the initial approach for comparative analysis with our proposed method.

4.2. Results and discussions

We have examined the various cases of camera activation while considering the single as well as multi-event occurrence scenarios.

4.2.1. Single event occurrence scenario

The effect of varying several parameters in case of single event occurrence scenario is studied in two ways i.e. without considering threshold and by considering the threshold.

Case 1. Without considering threshold values: We varied the number of scalars, number of cameras and depth of fields (DOFs) of cameras individually and observed their impact on the several performance metrics.

(i) *Effect of varying number of scalar sensors*

We varied the number of scalars and observed its effect on the number of cameras activated as shown in Fig. 2. We observed that with increase in the number of scalars, the number of cameras activated goes on increasing. This happens since with increase in number of scalars, more number of scalars detects the event occurrence, leading to activation of more number of cameras. However, after some time the number of cameras activated starts to decrease since with excess increase in number of scalars, overlapping FOV regions among cameras increases. As a result, more regions become common among the concerned cameras. Therefore, less number of cameras is activated. Moreover, since the number of cameras activated in the proposed approach is less, hence, the amount redundant data transmission is also less which is represented by reduced values of *redundancy ratio* in case of our approach as shown in Fig. 3(a). Further, with a rise in number of scalars the *coverage ratio* rises as shown in Fig. 3(b) and suddenly it falls due to random deployment of nodes. It is observed that the coverage ratio is more in the proposed approach than the initial approach [1]. This ensures that the actuated cameras cover more distinct portions of event region leading to improved coverage of event region. Moreover, due to activation of lesser number of cameras, the amount of energy expenditure for camera activation is found to be lowered in our method as portrayed in Fig. 4.

(ii) *Effect of varying number of camera sensors*

The result shown in the Fig. 5 reveals that with increase in number of cameras, the number of cameras activated goes on increasing since increase in number of cameras results in coverage of more portions of area of occurring event. However, at the value of No. of cameras = 170, the number of cameras activated suddenly decreases in the proposed approach due to random deployment of sensors. Further, the number of cameras activated in our approach is seen to be less than that of event boundary approach leading to minimized *redundancy ratio* in our case than the initial approach as shown in Fig. 6 (a). Similarly, with rise in number of cameras, the coverage ratio is observed to be more in our proposed approach as shown in Fig. 6(b). Moreover, Fig. 7 reveals that the energy consumption for camera activation is lesser in the proposed approach as compared to event boundary approach used in [1].

(iii) *Effect of varying depth of field (DOF) on number of cameras activated*

The results shown in Fig. 8 reveal that the number of cameras activated in the proposed approach is less than that of the

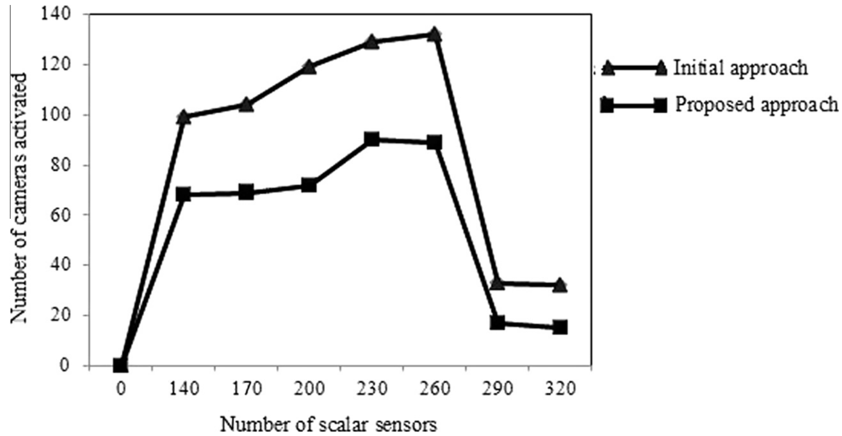

Figure 2 Number of Scalar sensors versus number of Cameras activated (without thresholds).

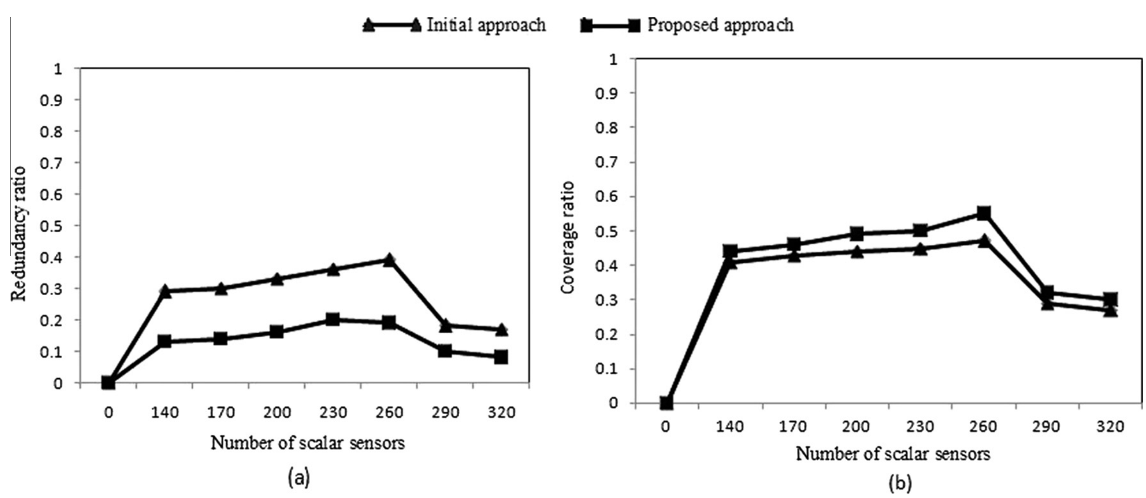

Figure 3 Number of Scalar sensors versus (a) Redundancy ratio (without thresholds) (b) Coverage ratio (without thresholds).

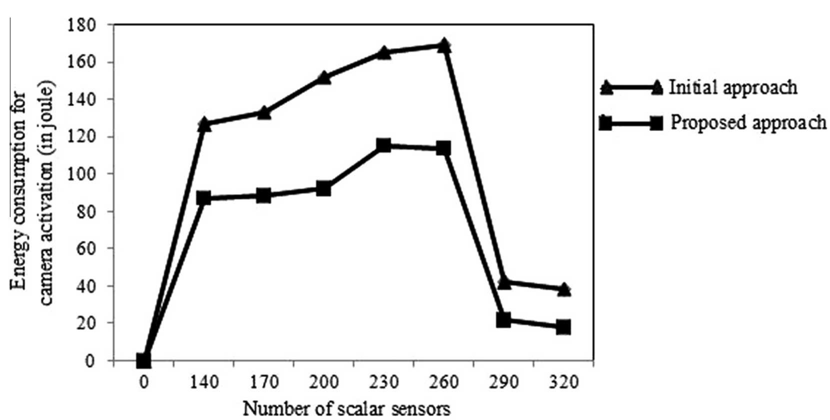

Figure 4 Number of Scalar sensors versus Energy consumption for camera activation (in joule) (without thresholds).

initial approach. Hence, the *redundancy ratio* is lowered in case of the proposed method as seen in Fig. 9(a). At the same time, the coverage ratio is also enhanced in our proposition as illustrated in Fig. 9(b). Besides, Fig. 10 depicts that the energy consumption for camera activation is lesser in our approach than event boundary approach [1].

Case 2. By considering threshold values: The threshold values are represented as *0.1*, *0.2*, *0.3*, *0.4*, *0.5* above the bars in Fig. 11–16. We have varied the parameters such as the number of scalars, number of cameras, and depth of fields (DOFs) of cameras individually and observed their effect on the number of cameras activated.

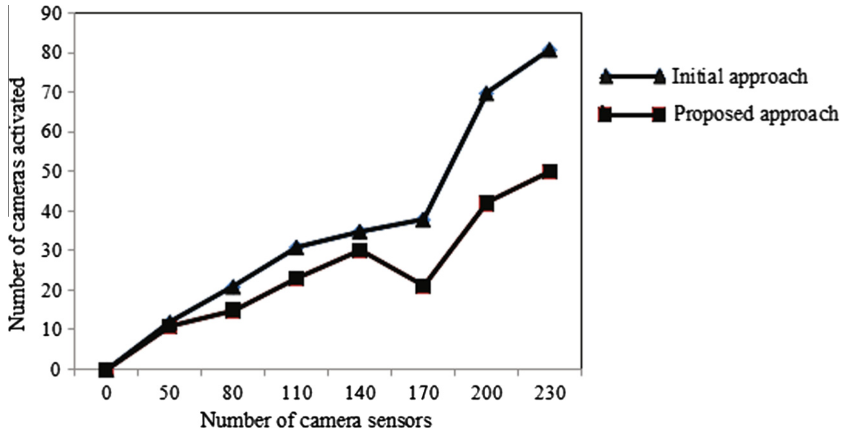

Figure 5 Number of Camera sensors versus number of Cameras activated (without thresholds).

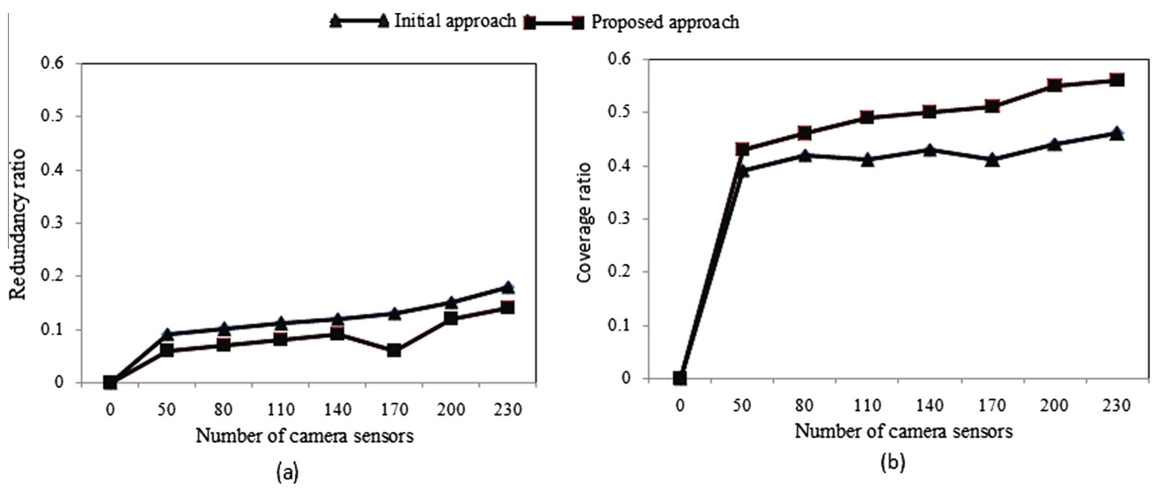

Figure 6 Number of Camera sensors versus (a) Redundancy ratio (without thresholds) (b) Coverage ratio (without thresholds).

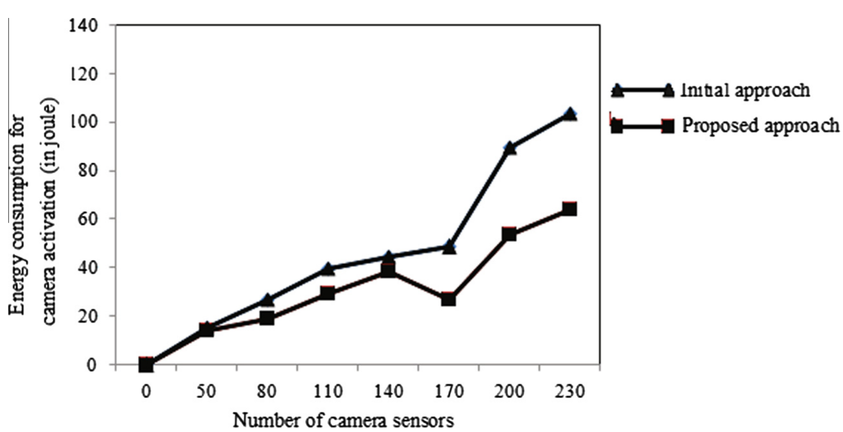

Figure 7 Number of Camera sensors versus Energy consumption for camera activation (in joule) (without thresholds).

Condition chosen for camera activation (While considering threshold value):

$$If\ ALPHA \leq (NEDS - UCNS)/NFC,\ then\ the\ camera\ is\ activated$$

Otherwise, the camera sensor is kept in turned off condition

where

ALPHA: Threshold value while considering camera activation.

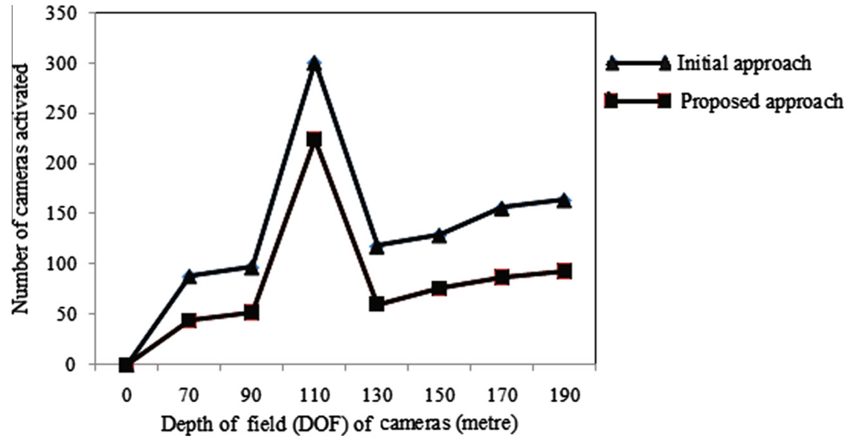

Figure 8 Depth of field (DOF) versus number of Cameras activated (without thresholds).

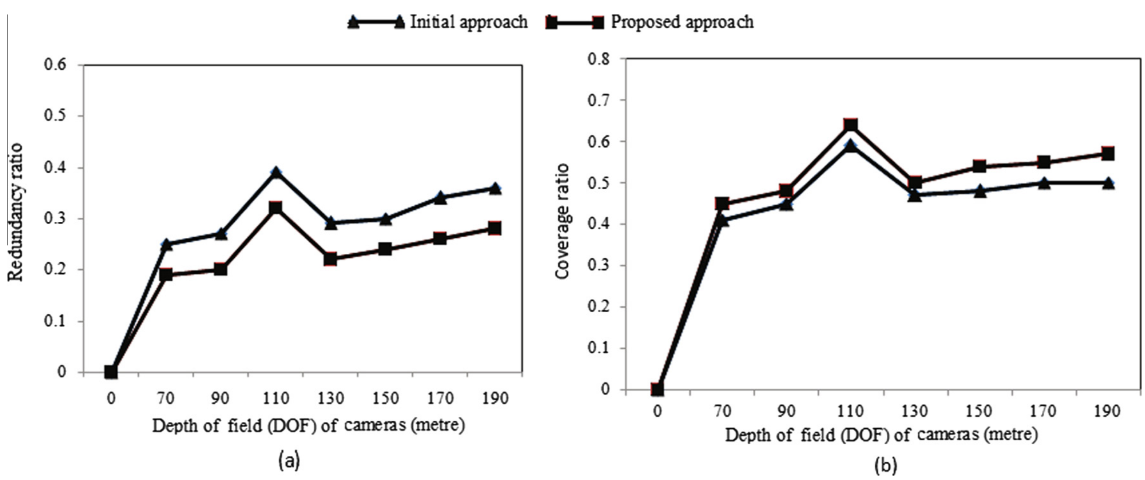

Figure 9 Depth of field (DOF) versus (a) Redundancy ratio (without thresholds) (b) Coverage ratio (without thresholds).

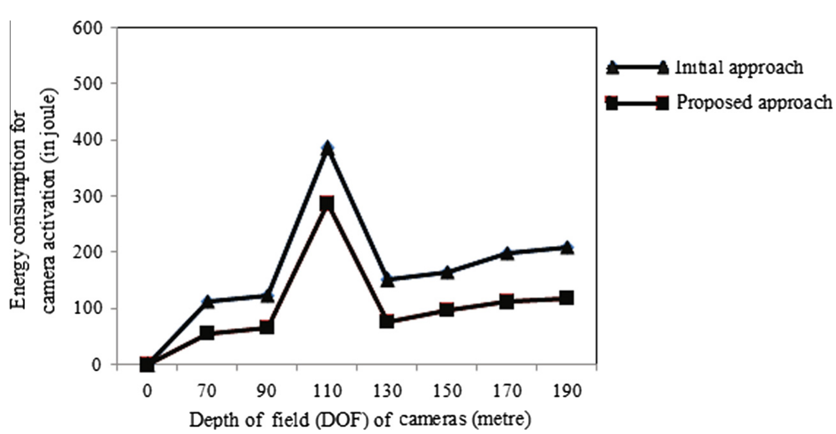

Figure 10 Depth of field (DOF) versus Energy consumption for camera activation (in joule) (without thresholds).

NEDS: Number of events detecting scalars present at event boundary region.
UCNS: Unique common neighbouring scalars.
NFC: Total number of scalars present within *FOVs* of cameras.

(i) *Effect of varying number of scalars on number of cameras actuated*

We varied the number of scalars and observed its effect on number of cameras activated in both initial approach and

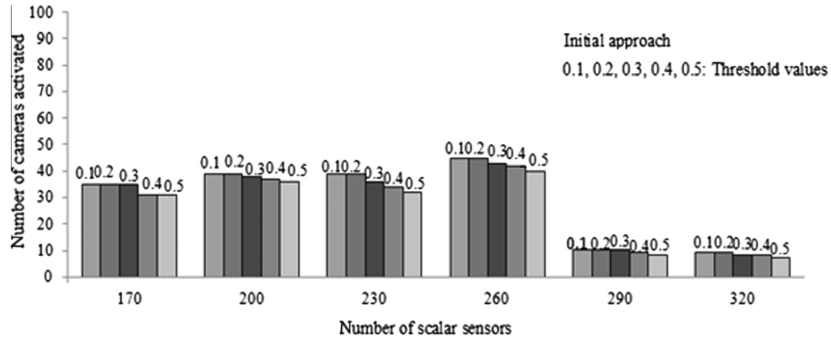

Figure 11 Number of Scalar sensors versus number of Cameras activated (initial approach).

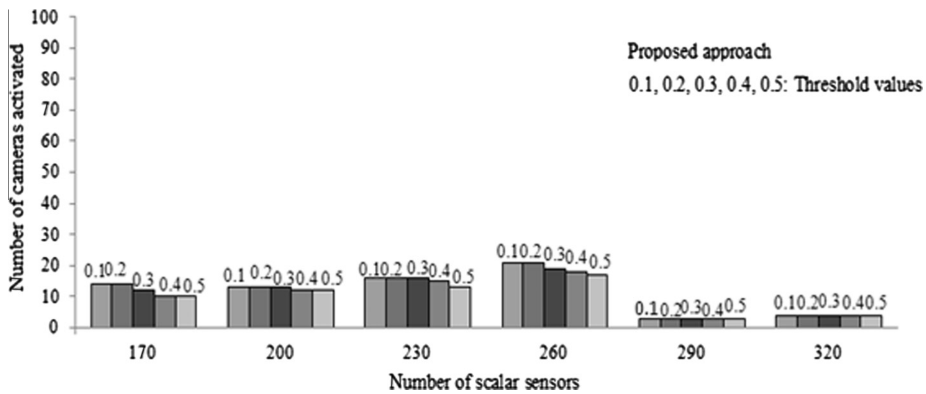

Figure 12 Number of Scalar sensors versus number of Cameras activated (proposed approach).

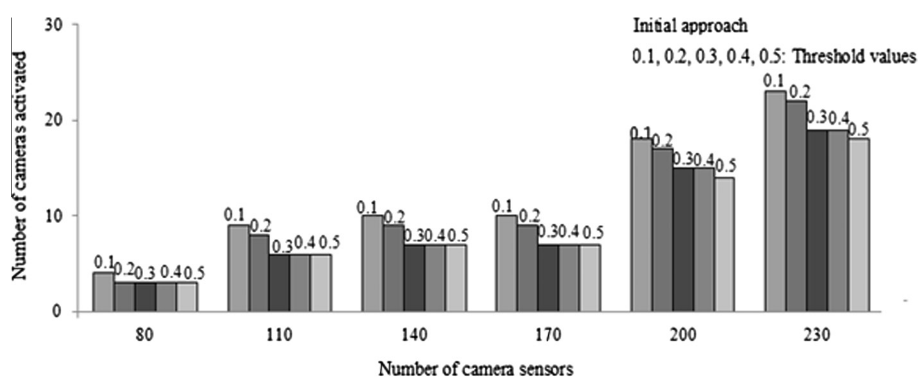

Figure 13 Number of Camera sensors versus number of Cameras activated (initial approach).

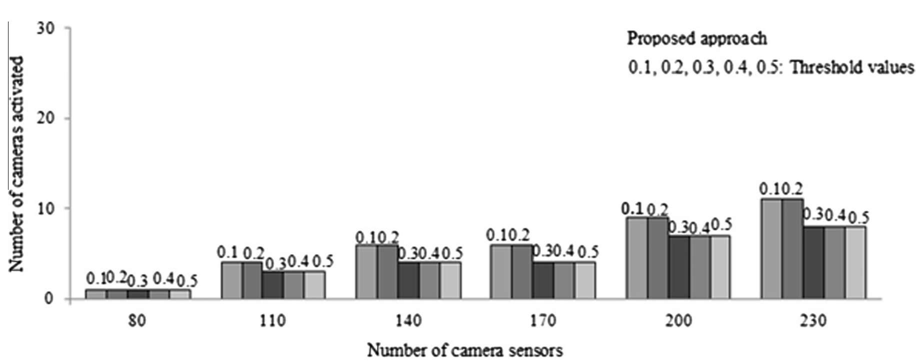

Figure 14 Number of Camera sensors versus number of Cameras activated (proposed approach).

Please cite this article in press as: Priyadarshini SBB, Panigrahi S, A novel approach using optimum camera actuation in event boundary detection method for redundant data minimization, Ain Shams Eng J (2016), http://dx.doi.org/10.1016/j.asej.2016.04.013

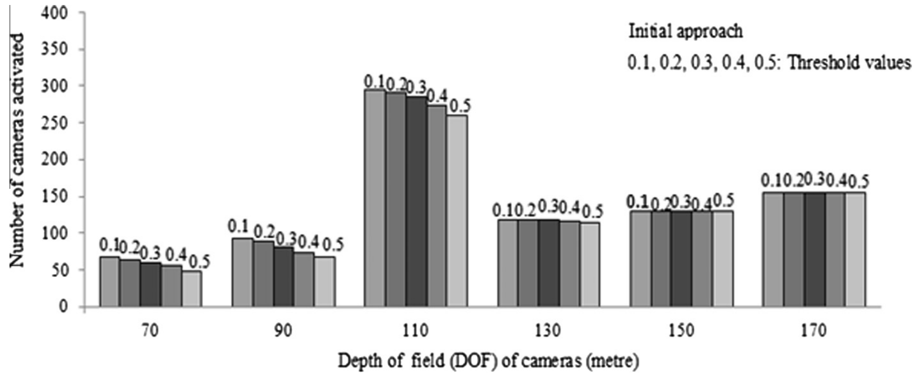

Figure 15 Depth of fields (DOFs) of camera sensors versus number of cameras activated (initial approach).

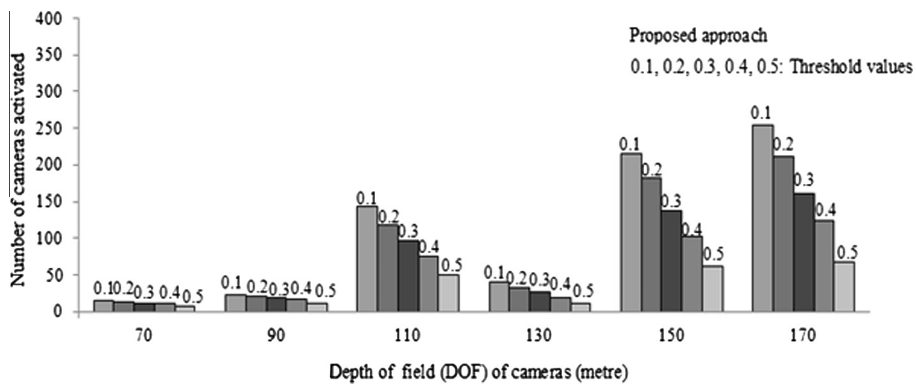

Figure 16 Depth of fields (DOFs) of Camera sensors versus number of Cameras activated (proposed approach).

Table 1 Effect of varying number of scalar sensors (*noss*) in multi-event scenario.

noss	*noca*		*cr*		*rr*		*Ecca (in joule)*	
	Initial approach	Proposed approach	Initial approach	Proposed approach	Initial approach	Proposed approach	Initial approach	Proposed approach
130	81	74	0.37	0.47	0.40	0.36	103.68	94.72
160	93	88	0.38	0.49	0.41	0.39	119.04	112.64
190	98	92	0.39	0.5	0.42	0.4	125.44	117.76
220	110	103	0.43	0.51	0.44	0.41	140.8	131.84
250	131	125	0.44	0.54	0.45	0.42	167.68	160

Table 2 Effect of varying number of camera sensors (*nocs*) in multi-event scenario.

nocs	*noca*		*cr*		*rr*		*Ecca (in joule)*	
	Initial approach	Proposed approach	Initial approach	Proposed approach	Initial approach	Proposed approach	Initial approach	Proposed approach
120	40	32	0.46	0.53	0.52	0.48	51.2	40.96
140	43	39	0.47	0.55	0.53	0.49	55.04	49.92
160	47	42	0.48	0.56	0.54	0.5	60.16	53.76
180	51	46	0.5	0.57	0.55	0.51	65.28	58.88
200	58	52	0.52	0.58	0.57	0.53	74.24	66.56

proposed approach as displayed in Figs. 11 and 12 respectively. It is evident from the figures that the number of cameras actuated is lowered in our method than the initial approach.

(ii) *Effect of varying number of camera sensors on number of cameras actuated*

The effect of varying number of cameras on number of cameras activated is portrayed in Figs. 13 and 14 for the initial approach and proposed approach respectively. The number of cameras actuated in proposed approach is lesser than initial one.

(iii) *Effect of varying depth of field (DOF) on number of cameras actuated*

Figs. 15 and 16 show the variation of *DOF* on the number of activated cameras and the number of cameras actuated in case of proposed method is found to be reduced.

4.2.2. Multi-event (object) occurrence scenario

We have varied the number of scalar sensors (*noss*) and number of cameras (*nocs*) individually and observed their effect on several parameters such as number of cameras activated (*noca*), redundancy ratio (*rr*), coverage ratio (*cr*) and energy consumption for camera activation (*ecca*) in case of multi-event occurrence scenario as depicted in Tables 1 and 2 respectively. While varying *noss* and *nocs*, it is observed that the *noca* values rise and are found to be lower in case of our proposed method. Further, the numerical results presented in both the tables establish that the *cr* value is increased, *rr* is reduced and *ecca* is minimized in case of our proposed method as compared to the event boundary method given in [1].

5. Conclusions and future work

In this paper, we have proposed a novel algorithm for optimum camera activation in event boundary detection method. In event boundary detection method, the cameras present outside the event boundary are unnecessarily activated. However, in the proposed method, we have overcome this problem of unnecessary camera activation by keeping the outer cameras in off condition by activation only the cameras present within the exact boundary. Extensive experimentation along with comparative analysis has been carried out for showing the effectiveness of the proposed algorithm in case of both single and multi-event scenarios. The reduced camera activation, less redundancy ratio, enhanced coverage ratio and minimized energy expenditure obtained from the experimental evaluation validate the superiority of the proposed approach over the initial approach.

We are also currently working on this aspect of dealing with obstacles that may be present in the monitored region, which can be modelled by placing lines that act as obstacles in the region under consideration. Further, we ensure our proposed algorithm does not fail completely even if obstacles are present due to random deployment of large number of cameras. If huge number of cameras is present and a particular camera fails to trap the image of any object due to the presence of obstacle, then any other neighbouring camera can carry out its task. However, it may be the case that the portions of area ensnared by both of them may vary, but complete event information loss never occurs.

Acknowledgements

The authors are highly grateful to the Department of Computer Science and Engineering, Veer Surendra Sai University of Technology (VSSUT), Burla, Sambalpur, India, for providing the required amenities and support for making this investigation successful.

References

[1] Newell Andrew, Akkaya Kemal. Distributed collaborative camera actuation for redundant data elimination in wireless multimedia sensor networks. Ad Hoc Networks 2011;9(4):514–27.
[2] Zorbas Dimitrios, Glynos Dimitris, Kotzanikolaou Panayiotis, Douligeris Christos. Solving coverage problems in wireless sensor networks using cover sets. Ad Hoc Networks 2010;8(4):400–15.
[3] Wang Jian, Niu Changyong, Shen Ruimin. Priority-based target coverage in directional sensor networks using a genetic algorithm I. Comput Math Appl 2009;57(11–12):1915–22.
[4] Ram S Sundhar, Manjunath D, Iyer Srikanth K, Yogeshwaran D. On the path coverage properties of random sensor networks. IEEE Trans Mobile Comput 2007;6(5):494–506.
[5] Cai Yanli, Lou Wei, Li Minglu, Li Xiang-Yang. Target-oriented scheduling in directional sensor networks. In: Proceedings of 26th IEEE international conference on computer communications (IEEE INFOCOM 2007), 6–12 May 2007, Barcelona. p. 1550–8.
[6] Girault Alain. Elimination of redundant messages with a two-pass static analysis algorithm. Parallel Comput 2002;28(3):433–53.
[7] Ghaddar Alia, Razafindralambo Tahiry, Tawbi Samar, Hijazi Abbas. Algorithm for data similarity measurements to reduce data redundancy in wireless sensor networks. In: IEEE international symposium on a world of wireless, mobile and multimedia networks (WoWMoM), June 14–June 17, 2010, Montreal, QC, Canada, ISBN 978-1-4244-7264-2.
[8] Toumpis S, Tassiulas T. Optimal deployment of large wireless sensor networks. IEEE Trans Inf Theory 2006;52(7):2935–53.
[9] Han X, Cao X, Lloyd E, Shen C. Deploying directional sensor networks with guaranteed connectivity and coverage. In: Proceedings of fifth annual IEEE communications society conference on sensor, mesh and ad hoc communications and networks. SECON; 2008. p. 153–60. http://dx.doi.org/10.1109/SAHCN.
[10] Al-Shalabi Ahmed Abdullah, Manaf Mazani. DKCS – an efficient dynamic K-coverage scheduling algorithm for wireless sensor networks. In: Proceedings of international symposium on telecommunication technologies (ISTT), 26–28 November, 2012, Kuala Lumpur. IEEE; 2012. p. 94–9.

Sushree Bibhuprada B. Priyadarshini received her B.Tech. Degree in Information Technology (IT) from Gandhi Institute for Technology (GIFT), Bhubaneswar, in 2011. She has done her M.Tech. from Institute of Technical Education and Research (ITER), Bhubaneswar, in Computer Science and Data Processing (CSDP) branch. Currently, she is pursuing her Ph.D. in Computer Science and Engineering (CSE) at Veer Surendra Sai University of Technology (VSSUT), Burla. Her area of interest is Wireless Sensor Networks.

Dr. Suvasini Panigrahi has done her Ph.D. from IIT Kharagpur in 2010 after completing her B.Tech. and M.Tech. Degrees in Comp. Sc. & Engg. under Utkal University in 2002 and 2004 respectively. Currently, she is working as Associate Prof. in the Department of Comp. Sc. & Engg. and IT at Veer Surendra Sai University of Technology (VSSUT), Burla. Her area of interest includes Database Security and Wireless Sensor Networks.