21

Power Management in Sensing Subsystem of Wireless Multimedia Sensor Networks

Mohammad Alaei and Jose Maria Barcelo-Ordinas
Computer Architecture Department, Universitat Politecnica de Catalunya, Barcelona
Spain

1. Introduction

A wireless sensor network consists of sensor nodes deployed over a geographical area for monitoring physical phenomena like temperature, humidity, vibrations, seismic events, and so on. Typically, a sensor node is a tiny device that includes three basic components: a sensing subsystem for data acquisition from the physical surrounding environment, a processing subsystem for local data processing and storage, and a wireless communication subsystem for data transmission. In addition, a power source supplies the energy needed by the device to perform the programmed task. This power source often consists of a battery with a limited energy budget. In addition, it is usually impossible or inconvenient to recharge the battery, because nodes are deployed in a hostile or unpractical environment. On the other hand, the sensor network should have a lifetime long enough to fulfill the application requirements. Accordingly, energy conservation in nodes and maximization of network lifetime are commonly recognized as a key challenge in the design and implementation of WSNs.

Experimental measurements have shown that generally data transmission is very expensive in terms of energy consumption, while data processing consumes significantly less (Raghunathan et al., 2002). The energy cost of transmitting a single bit of information is approximately the same as that needed for processing a thousand operations in a typical sensor node (Pottie & Kaiser, 2000). The energy consumption of the sensing subsystem depends on the specific sensor type. In some cases of scalar sensors, it is negligible with respect to the energy consumed by the processing and, above all, the communication subsystems. In other cases, the energy expenditure for data sensing may be comparable to, or even greater (in the case of multimedia sensing) than the energy needed for data transmission. In general, energy-saving techniques focus on two subsystems: the communication subsystem (i.e., energy management is taken into account in the operations of each single node, as well as in the design of networking protocols), and the sensing subsystem (i.e., techniques are used to reduce the amount or frequency of energy-expensive samples).

1.1 Power consumption in sensing subsystem

In fact, the energy consumption of the sensing subsystem not only may be relevant, but it can also be greater than the energy consumption of the radio or even greater than the energy consumption of the rest of the sensor node (Alippi et al., 2007). This can be due to many different factors (Raghunathan et al., 2006):

- Power hungry transducers. Some sensors intrinsically require high power resources to perform their sampling task. For example, sensing arrays such as CCDs or multimedia sensors (Akyildiz et al., 2007) such as CMOS image sensors generally require a lot of power. Also chemical or biological sensors (Diamond, 2006) can be power hungry as well.
- Long acquisition time. The acquisition time may be in the order of hundreds of milliseconds or even seconds, especially in the case of multimedia sensors. Hence the energy consumed by the sensing subsystem may be high, even if the sensor power consumption is moderate. In this case reducing communications may be not enough, but energy conservation schemes have to actually reduce the number of acquisitions (i.e. data samples). It should also be pointed out that energy-efficient data acquisition techniques are not exclusively aimed at reducing the energy consumption of the sensing subsystem. By reducing the data sampled by source nodes, they decrease the number of communications as well. Actually, many energy-efficient data-acquisition techniques have been conceived for minimizing the radio energy consumption, under the assumption that the sensor consumption is negligible.
- Power hungry A/D converters. Sensors like acoustic and seismic transducers generally require high-rate and high-resolution A/D converters. The power consumption of the converters can account for the most significant power consumption of the sensing subsystem, as in (Schott et al., 2005).

1.2 Multimedia sensing subsystem

One of the main differences between multimedia sensor networks and other types of sensor networks lies in the nature of how the image sensors perceive information from the environment. Most scalar sensors provide measurements as 1-dimensional data signals. However, image sensors are composed of a large number of photosensitive cells. One measurement of the image sensor provides a 2-dimensional set of data points, which we see as an image. The additional dimensionality of the data set results in richer information content as well as in a higher complexity of data processing and analysis. In addition, a camera's sensing model is inherently different from the sensing model of any other type of sensor. Typically, a scalar sensor collects data from its vicinity, as determined by its sensing range. Multimedia nodes are characterized by a directional sensing model, called Field of View (FoV, see Figure 1), and can capture images of distant/vicinal objects/scenes within its FoV from a certain direction. The object covered by the camera can be distant from the camera and the captured images will depend on the relative positions and orientation of the cameras towards the observed object (Soro & Heinzelman, 2005; Tezcan & Wang, 2008; Adriaens et al., 2006). Because of non-coincidence between neighborhood and sensed region by multimedia nodes, coverage-based techniques in WSN do not satisfy WMSN requirements.

Accordingly, the amount of power consumed in the sensing subsystem of a multimedia sensor node is considerably more than of a scalar ordinary sensor. For example, a temperature sensor (texas instrument, 2011) as a scalar sensor consumes 6µW for sensing the environment. To have a view of multimedia sensors power consumption, table 1 shows the power consumed by four classes of cameras that are available today either as prototypes or as commercial products. At the lowest end of the spectrum is tiny Cyclops (Rahimi et al., 2005) that consumes a mere 46mW and can capture low resolution video. CMU-Cams (Rowe et al., 2002) are cell-phone class cameras with on-board processing for motion detection, histogram computation, etc. At the high-end, web-cams can capture high-resolution video at full frame rate while consuming 200mW, whereas Pan-Tilt-Zoom cameras are re-targetable sensors that produce high quality video while consuming 1W. It is noticeable that the mentioned power amounts are the power consumed by the camera sensors without considering the power consumed by the host motes, see (Tavli et al, 2011) for a survey of visual network platforms.

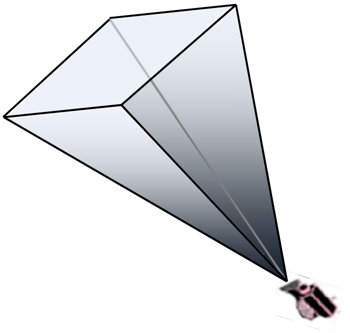


Fig. 1. The Field of View (FoV) of a multimedia sensor node.

Multimedia Sensor	Power of image capturing	Capability in image capturing
Cyclops	42 mW	Fixed angle lens, 352×288 at 10 fps
CMU-Cam	200 mW	Fixed angle lens, 352×288 up to 60 fps
Web-Cam	200 mW	Auto focus lens, 640×480 at 30 fps
High-end PTZ Camera	1 W	Pan-tilt-zoom lens, 1024×768 up to 30fps

Table 1. Power consumption and capabilities of four classes of camera sensors.

On the other hand, given the large amount of data generated by the multimedia nodes, both processing and transmitting image data are quite costly in terms of energy, much more so than for other types of sensor networks. Furthermore, visual sensor networks require large bandwidth for transmitting image data. Thus both energy and bandwidth are even more constrained than in other types of wireless sensor networks.

In this chapter, we describe a power efficient mechanism for managing the sensing subsystem of multimedia sensor nodes for surveillance in WMSNs. For this purpose, the deployed multimedia nodes are clustered according to their common covering regions and the clusters are managed to schedule the members to collaboratively survey the sensing area in a duty-cycled manner. With avoiding acquisition of redundant and correlated data, not only the sensing subsystem of nodes save its energy, but also the transmission and processing subsystems meet an optimized amount of data to be transmitted/processed and thus can conserve their residual energy. Therefore, the network lifetime is considerably prolonged.

The chapter is organized as follows. In section 2 we present an overview of work related to sensor management and scheduling policies. A surveillance mechanism with its details in grouping, management and scheduling multimedia nodes to be energy efficient is explained in section 3. Finally, the future work and conclusions are derived.

2. Sensor management and scheduling policies

In redundantly deployed multimedia sensor networks a subset of cameras can perform continuous monitoring and provide information with a desired quality. This subset of active cameras can be changed over time, which enables balancing of the cameras energy consumption, while spreading the monitoring task among the cameras. In such a scenario the decision about the camera nodes activity and the duration of their activity is based on sensor management policies. *Sensor management policies* define the selection and scheduling (that determines the activity duration) of the camera nodes activity in such a way that the visual information from selected cameras satisfies the application specified requirements while the use of camera resources is minimized. Various quality metrics are used in the evaluation of sensor management policies, such as the energy-efficiency of the selection method or the quality of the gathered image data from the selected cameras. In addition, camera management policies are directed by the application; for example, target tracking usually requires selection of cameras that cover only a part of the scene that contains the non-occluded object, while monitoring of large areas requires the selection of cameras with the largest combined FoV. While energy-efficient organization of camera nodes is oftentimes addressed by camera management policies, the quality of the data produced by the network is the main concern of the application.

The problem of finding the best camera candidates is investigated in (Soro & Heinzelman, 2007). In this work, the authors propose several cost metrics for the selection of a set of camera nodes that provide images used for reconstructing a view from a user-specified view point. Two types of metrics are considered: coverage aware cost metrics and quality-aware cost metrics. The *coverage-aware cost metrics* consider the remaining energy of the camera nodes and the coverage of the indoor space, and favor the selection of the cameras with higher remaining energy and more redundant coverage. The *quality-aware cost metrics* favor the selection of the cameras that provide images from a similar view point as the user's view point. Thus, these camera selection methods provide a trade-off between network lifetime and the quality of the reconstructed images.

Monitoring of large areas (such as parking lots, public areas, large stores, etc.) requires complete coverage of the area at every point in time. Such an application is analyzed in (Dagher et al., 2006), where the authors provide an optimal strategy for allocating parts of the monitored region to the cameras while maximizing the lifetime of the camera nodes. The optimal fractions of regions covered by every camera are found in a centralized way at the base station. The cameras use JPEG2000 to encode the allocated region such that the cost per bit transmission is reduced according to the fraction received from the base station.

Oftentimes, the quality of a reconstructed view from a set of selected cameras is used as a criterion for the evaluation of camera selection policies. In the work (Park et al., 2006) distributed look-up tables are used to rank the cameras according to how well they image a specific location, and based on this, they choose the best candidates that provide images of the desired location. Their selection criterion is based on the fact that the error in the captured image increases as the object gets further away from the center of the viewing frustum. Thus, they divide the frustum of each camera into smaller unit volumes (subfrustums). Then, based on the Euclidian distance of each 3D point to the centers of subfrustums that contain this 3D point, they sort the cameras and find the most favorable camera that contains this point in its field of view. The look-up table entries for each 3D location are propagated through the network in order to build a sorted list of favorable cameras. Thus, camera selection is based exclusively on the quality of the image data provided by the selected cameras, while the resource constraints are not considered.

In order to reduce the energy consumption of cameras, the work (Zamora & Marculescu, 2007) explores distributed power management of camera nodes based on coordinated node wake-ups. The proposed policy assumes that each camera node is awake for a certain period of time, after which the camera node decides whether it should enter the low-power state based on the timeout statuses of its neighboring nodes. Alternatively, camera nodes can decide whether to enter the low-power state based on voting from other neighboring cameras.

Selection of the best cameras for target tracking has been discussed often (Pahalawatta et al., 2004; Ercan et al., 2006). Pahalawatta et al. present a camera selection method for target tracking applications used in energy-constrained visual sensor networks. The camera nodes are selected by minimizing an information utility function (obtained as the uncertainty of the estimated posterior distribution of a target) subject to energy constraints. However, the information obtained from the selected cameras can be lost in the case of object occlusions. This occlusion problem is further discussed by Ercan et al. where they propose a method for camera selection in the case when the tracked object becomes occluded by static or moving occluders. Finding the best camera set for object tracking involves minimizing the MSE of the object position's estimates. Such a greedy heuristic for camera selection shows results close to optimal and outperforms naive heuristics, such as selection of the closest set of cameras to the target, or uniformly spaced cameras. The authors here assume that some information about the scene is known in advance, such as the positions of static occluders, and the object and dynamic occluders prior probabilities for location estimates.

As a conclusion, in multimedia sensor networks, sensor management policies are needed to assure balance between the opposite requirements imposed by the wireless networking and vision processing tasks. While reducing energy consumption by limiting data transmissions is the primary challenge of energy-constrained visual sensor networks, the quality of the image data and application, QoS, improve as the network provides more data. In such an environment, the optimization methods for sensor management developed for wireless sensor networks are hard to directly apply to multimedia sensor networks. Such sensor management policies usually do not consider the event-driven nature of multimedia sensor networks, nor do they consider the unpredictability of data traffic caused by a monitoring procedure. Thus, more research is needed to further explore sensor management for multimedia sensor networks. Since sensor management policies depend on the underlying networking policies and vision processing, future research lies in the intersection of finding the best trade-offs between these two aspects of visual sensor networks. Additional work is needed to compare the performance of different camera node scheduling sensor policies, including asynchronous (where every camera follows its own on-off schedule) and synchronous (where cameras are divided into different sets, so that in each moment one set of cameras is active) policies. From an application perspective, it would be interesting to explore sensor management policies for supporting multiple applications utilizing a single visual sensor network.

The presented mechanism in the following section groups multimedia nodes in clusters based on their common sensing region of the whole deployment region. The clusters monitor the environment independently but in each cluster the members collaborate in data acquisition in an intermittent manner. The scheduling and activity times in each cluster are determined based on the cluster population and the scale of overlapping between FoV of cluster members. So, the data transmissions are not limited in this kind of sensor management but the volume of sensed data is reduced by management in only sensing subsystem and applying coordination among cluster members to optimize capturing image times and to avoid redundant sensing of the same data in the overlapped FoVs. On the other hand, the sensing region is divided between clusters and each cluster monitors its domain with its exclusive frequency and member scheduling. Thus, clusters are not synchronized for sensing the region whiles each point of the sensing region is monitored frequently according to the number of nodes that cover that point by their sensing subsystem.

3. The surveillance mechanism

3.1 Preliminary

We assume wireless sensor nodes with fixed lenses providing a θ angle FoV, densely deployed in a random manner. The assumption of fixed lenses is based on the current WMSN platforms (Tavli et al, 2011). Almost all of them (SensEye, MicrelEye, CITRIC, Panoptes, Meerkats) (Kulkarni et al., 2005; Kerhat et al., 2007; Chen et al., 2008; Feng et al., 2005; Margi et al., 2006) have fixed lenses and only high powered PTZ cameras have movement capabilities. We consider a monitor area with N wireless multimedia sensors, represented by the set $S = \{S_1, S_2, ..., S_N\}$ randomly deployed. Each sensor node is equipped to learn its location coordinates and orientation information via any lightweight localization technique for wireless sensor networks. It is not the purpose of this chapter to define mechanisms to find this location. Without loss of generality, let us assume that nodes in the set S belong to a single-tier network or the same tier of a multitier architecture.

Our policy in order to applying collaboration among multimedia sensor nodes in the surveillance mechanism is clustering the network nodes based on their similarity in sensing the environment. The criterion applied in this purpose is the clustering scale of FoVs of nodes. The nodes having a large region of common area in their FoV, have a similar view of the sensing area then can cooperate in a established group, (Alaei & Barcelo, 2010).

3.2 Cluster formation and cluster membership

Now, let us consider the set $S = \{S_1, S_2, ..., S_N\}$ of wireless multimedia nodes belonging to the same tier of a network randomly deployed. The cluster formation algorithm is executed in a centralized manner by the sink after deploying the network. The main reasons in choosing a central architecture are the following: (i) for a distributed architecture, each node should notify to the rest of the nodes about its location A_i and its orientation α_i (i = 1,…,N). In a centralized architecture the nodes should notify to the sink their location and orientation. Note that this notification can be done using any energy efficient sensor routing protocol and only is necessary at bootstrap phase. All phases of the clustering algorithm are executed only one time, right after node deployment. (ii) In many WSN applications, the sink has ample resources (storage, power supply, communication and computation) availability and capacity which make it suitable to play such a role. (iii) Collecting information by a sink node is more power efficient compared to spreading this information to each and every other node within the network. (iv) Having the global view of the network at the sink node facilitates provision algorithms for closer-to-optimal cluster determination; the global knowledge can be updated at the sink when new nodes are added or some nodes die. Such maintenance tasks can be regarded as a normal routine for the sink. (v) Finally, using a centralized scheme can relieve processing load from the sensors in the field and help in extending the overall network lifetime by reducing energy consumption at individual nodes. The following phases are performed to establish and form clusters, (Figure 2):

- *Bootstrap*: At node bootstrap, each sensor $\{S_i, i = 1,\ldots,N\}$ transmits its position (x_i,y_i) and orientation α_i to the sink. To accomplish this step any efficient sensor routing algorithm can be used. Thus, the clustering algorithm is not bound to how the sink receives this information. If there is an un-connected node in the network, it cannot announce itself and thus will not be considered in the algorithm.
- *Cluster Formation*: (i) Initially, the sink creates an empty cluster associated with an un-clustered multimedia node of S. Thus, that node will be clustered as the first member (*i.e.*, Cluster-Head (CH)) of the established cluster. (ii) Then, the sink finds the qualified un-clustered nodes for joining to the CH by computing the area of overlapped polygons of their FoV. From position and orientation of nodes, the sink computes the overlapped region between each un-clustered multimedia node and the CH of the established cluster. For calculating the FoV overlapping area of two nodes, we first survey the intersection of their FoVs. Second, if they intersect each other, we find the intersection region and at last, compute the area of the polygon. For this purpose, in the first step, we define the equations of the sides of FoVs using the vertex coordinates. Then, the intersection of each side of each FoV to all sides of the other is calculated. A decomposition approach is used for calculating the area of the overlapping region of FoVs. If the computed overlapped region is equal or greater than the threshold considered as the C*lustering Scale* (γ) -the minimum region that has to be overlapped between two node FoVs to be grouped in a cluster-, the un-clustered node will be clustered as a member of the established cluster. (iii) When no more nodes can be added to the cluster, the sink takes a new un-clustered node, begins a new cluster and goes to step (ii).
- *Membership notification*: we assume that the sink uses any energy-efficient sensor routing algorithm to notify to each first-member of every cluster about its cluster-ID and what are the members of the cluster. Then, each first-member sends a packet to the members of his cluster notifying them about the cluster which they belong to.

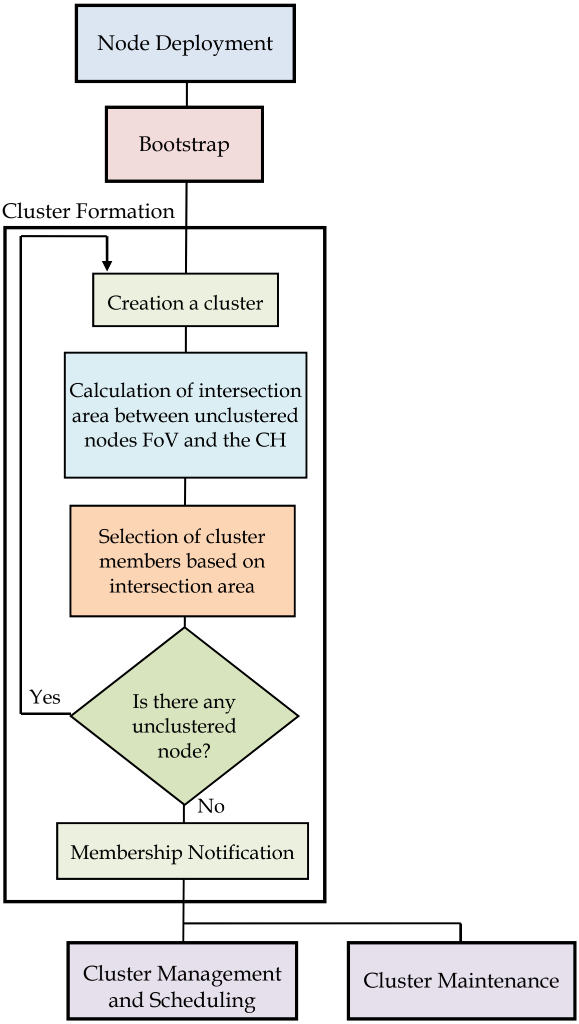

Node Deployment
Bootstrap
Cluster Formation
Creation a cluster
Calculation of intersection area between unclustered nodes FoV and the CH
Selection of cluster members based on intersection area
Is there any unclustered node?
Yes
No
Membership Notification
Cluster Management and Scheduling
Cluster Maintenance

Fig. 2. Clustering Procedure.

The algorithm is executed by the sink once upon deployment and thus all nodes will become clustered. If a node joins to the network hereinafter, it has to send its position and orientation to the sink for announcing itself as a new node. The sink computes the FoV of the new node and finds the first cluster that can accept it as a new member. For this purpose, the sink computes the overlapping regions between FoV of the new node and the CH of each cluster and checks whether he is satisfying the cluster membership test. Then, the sink sends a message to the CH in order that this node re-organizes the cluster with the new member. Depending on the application, this notification may suppose a new reconfiguration in the monitoring task (*i.e.*, a new duty-cycle period). On the other hand, each node periodically sends a Hello message to the CH notifying its current residual energy. When a node dies, the CH will notify the rest of the members about the new cluster set and will reconfigure any parameter related to the cluster. The CH also periodically compares the residual energy of cluster members and its residual energy to select the new CH with the maximum residual energy in the cluster. If the CH decides to entrust CH role to another cluster member, notifies to the cluster members about the new CH. Note that the beaconing among cluster members implies low overhead since clusters have few nodes and hello periods can be on the order of duty-cycle sensing periods.

3.2.1 Intra-cluster collaboration

Let us see the potential of cooperative node monitoring in clusters in terms of sensor area coverage. We define the Maximum Cluster Coverage Domain (MCCD) parameter for a cluster as the maximum monitoring area which is covered by that cluster. Since each cluster is established considering the clustering scale equal to γ, the MCCD can be computed as follows (C_{size} is the size of the cluster):

$$MCCD = \gamma \cdot A_{FoV} + (1-\gamma) \cdot A_{FoV} \cdot C_{size} = (C_{size} - \gamma \cdot (C_{size} - 1)) \cdot A_{FoV} = \beta \cdot A_{FoV} \quad (1)$$

where:

$$1 \le \beta = C_{size} - \gamma \cdot (C_{size} - 1) \quad (2)$$

The effective cluster covering domain can be inferior to the MCCD calculated by Equation (1) since some nodes can overlap more than the region determined by γ. Since MCCD gives us an upper bound on the area covered by the cluster, using MCCD will allow us worst-case dimensioning. Factor β represents the increment of area that the cluster senses with respect to an individual sensor. When each node of a cluster obtains an image from its FoV, a part of the related MCCD with a ratio at least equal to 1/β respect to the MCCD is captured whereas this part includes overlapped areas of other nodes in the cluster. Sensing the environment by each member delivers information not only from the FoV of the active node but also from some overlapped parts of other nodes in the same cluster: at least $\gamma \cdot A_{FoV}$ of the area is common to the first-member and more than 1/β of the MCCD is monitored. For example, in a cluster consisting of just 2 members, assuming a clustering scale of γ = 0.5, the MCCD is $1.5 \cdot A_{FoV}$. Thus, when each of the two members of the cluster is activated and monitors the environment, an area of one FoV is captured that is at least 2/3 of the whole MCCD of the cluster. Consequently, scheduling and coordination among members in order to sense the field in a collaborative manner may yield a gain in energy saving and performance efficiency even with a low number of members in the cluster.

3.2.2 Cluster formation evaluation

All sensor nodes have been configured with a FoV vertex angle of θ = 60° and R_S of 20 m. A sensing field spanning an area of 120m × 120m has been used. Sensor densities were varied to study the cluster formation from sparse to dense random deployments. Figures illustrate the average results of 50 independent running tests whereas each test corresponds to a different random deployment. Once a random deployment is defined, cluster formation is obtained from node location, angle of orientation and FoVs of nodes, using the described method whose complexity is O(N.logN). Furthermore, as it was mentioned before, each node sends a packet to the sink in the bootstrap phase, then the sink notifies each CH via one packet his membership set for that cluster (phase 3) and then the CHs notify cluster nodes about their cluster membership and any related parameter. Thus, the average overhead of the algorithm is forwarding N packets from the nodes to the sink and forwarding N_C packets from the sink to first-members and forwarding $N_C \cdot (\mu_{Csize}-1)$ packets from CHs to cluster nodes; where N is the number of nodes, N_C is the average number of clusters and μ_{Csize} is the average cluster size. So the total overhead will be: $N + N_C + N_C \cdot (\mu_{Csize}-1)$ packets. The maintenance overhead is $N_C \cdot (\mu_{Csize}-1)$ beacons every keep-alive period, where the keep-alive period can be a multiple of the sensing duty-cycle period.

3.2.2.1 Number of clusters and cluster-size

The average number of clusters, μ_{NC}, and the average cluster-size (μ_{Csize}) in a tier/network for different node densities with several clustering scales are shown in Figures 3 and 4. Increasing the node density does not only cause an increment in the number of clusters but also yields more overlapping areas among FoVs and thus raises the cluster-size. However, the clustering scale (γ) also impacts in the cluster membership selection process. The clustering scale determines the minimum region that is required to be overlapped between the FoV of each node belonging to a given cluster and the FoV of the CH of that cluster. So, γ determines the minimum intersection part of FoV of each member with the CH of an established cluster. Lower clustering scales obligate less overlapping areas for cluster membership and increase the domain covered by a given cluster since more nodes will be conforming to the membership rule. Increasing the clustering scale restricts node membership because of higher required overlapping areas between FoVs of nodes. Thus, higher clustering scales result in lower cluster-sizes, less MCCD and thus higher number of clusters.

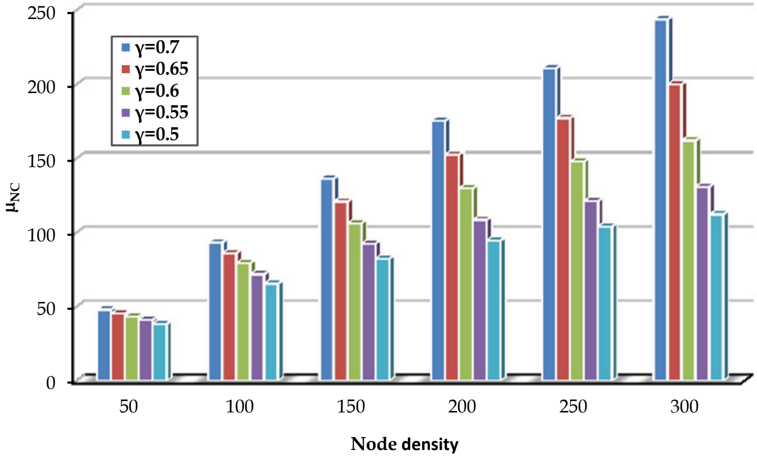


Fig. 3. Average number of established clusters.

Sparse networks have low average cluster-size, μ_{Csize}, because sparse deployments result in low overlapping areas. Moreover, high values of γ also will produce low μ_{Csize}. The result will be lower potential for node coordination. On the other hand, dense wireless multimedia sensor networks can particularly benefit from higher cluster sizes and thus more potential for node coordination.

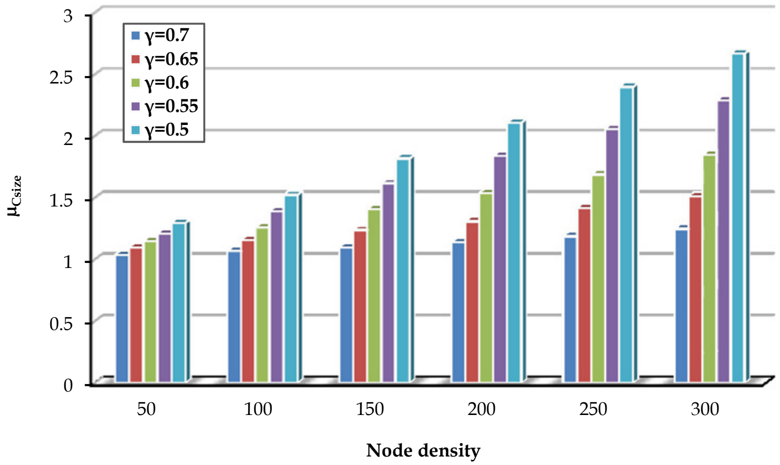


Fig. 4. Average size of established clusters.

Finally, Figure 5 shows the cumulative probability function for the cluster-size in the network for different node densities assuming a clustering scale of γ = 0.5. For example, in a network consisting of 250 nodes, 28% of clusters have a single member which does not have enough overlapping with others to satisfy the clustering scale, 32% of clusters have a cluster size of 2, 21% of 3, 12% of 4 and 7% of them consisting of more than four members.

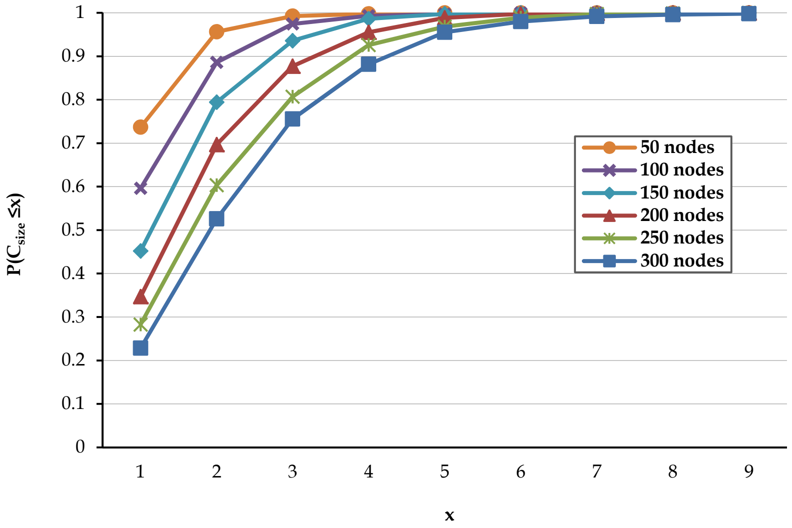


Fig. 5. The cluster size cumulative distribution function (γ = 0.5).

3.2.2.2 Coverage

Figure 6 illustrates the percentage of area that is covered by the random deployment in terms of node density. As it is shown in the figure, for covering 95% of the area, a dense deployment of 300 nodes is required. As the figure shows, the rate of increment of the covered area for low node densities is faster than for high node densities. This indicates that after a new node is added in a dense deployment, low new coverage area is obtained.

For example, the first 100 nodes cover 75% of the field, but the next 100 nodes will only cover 15% of new area. The conclusion is that dense networks are able to cover high areas at the cost of high overlapping and sensing redundancy, but this overlapping can be used for improving reliability if nodes belonging to the same cluster work in a coordinated manner. Furthermore, the existence of obstacles produces a reduction of the sensing area because of FoV occlusion effect, (Tezcan & Wang, 2008). So, employing dense networks of low-cost, low-resolution and low-power multimedia sensor nodes instead of sparse networks of high-power, high-resolution sensors (*e.g.*, PTZ) will be more beneficial.

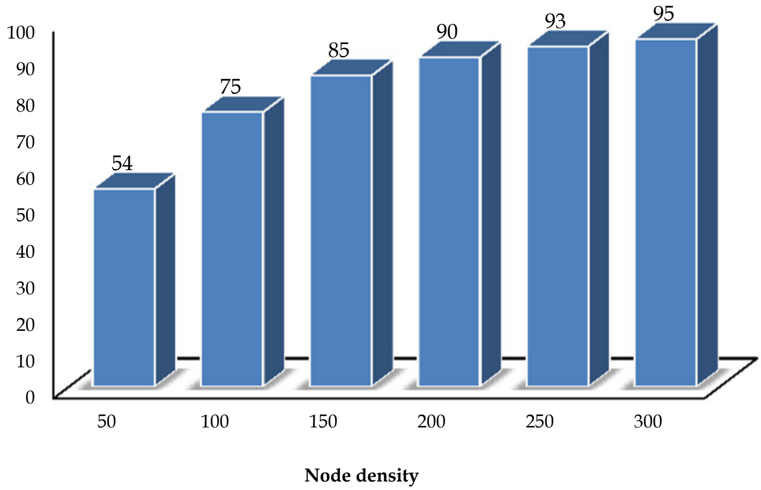


Fig. 6. Percentage of the covered area with respect to the whole deployment area.

Applications that are interested in multiple views will also benefit from this situation, since there will be several nodes monitoring the same area from several perspectives. Applications that are interested in detecting objects and are not interested in having an instantaneous multiple-view of the object may benefit from collaborative node processing in terms of energy savings. For the first set of applications clustering of nodes may serves as an indicator of triggering simultaneous multi-perspective pictures. For the second set of applications, clustering may serve as a baseline framework for collaborative node scheduling avoiding redundant sensing and processing and thus increasing network lifetime. Other applications that are interested in correlated data (*e.g.*, Distributed Video Coding, DVC) may use clustering in order to exploit multi-view correlations to build joint encoders (Pereira et al., 2008).

3.3 Cooperative node selection and scheduling

In monitoring mechanisms, usually cameras should perform duty-cycled monitoring over the area that they sense. That means that every T (Figure 7.a) seconds the sensors in the monitored area will awake and monitor the area. This is the situation for a planned network in which every sensor is placed in such a position that there is no overlapping among sensors. Nevertheless, this duty-cycle scheduling will produce high power consumption in those situations in which there are overlapping sensors, since camera nodes with overlapping areas do not cooperate to sense the area and thus they redundantly monitor the area.

In this section, we explain a cooperative mechanism based on the clustering method that coordinates nodes belonging to the same cluster to work in a collaborative manner to monitor the sensing area. The objective of this mechanism is to increase power conservation by avoiding similar sensing and redundant processing at the same time. Also, collaborative sensing by nodes that have FoVs intersecting each other yields to more reliability: cluster members will monitor the region sequentially and if a moving object is not detected in one image capturing, it will be in the vicinal FoVs at the next capturing times. Thus, the other members in the same cluster may detect the object.

Let us divide the environment in domains covered by clusters of nodes (MCCD, Section 3.2). All clusters concurrently sense their domains. In each cluster, members are awakened sequentially in an intermittent manner by the CH with a time interval related to the cluster-size and the scale of clustering (see Figure 7.b); (*i.e.*, $T_{interval}$ is the time between awakening two consecutive members of a cluster). In this way, each node of a given cluster periodically participates in capturing an image from its unique perspective and surveillance the environment and finally sleeps again with a cluster-based period called T_p. Formulas for these periods are derived in Section 3.3.1.

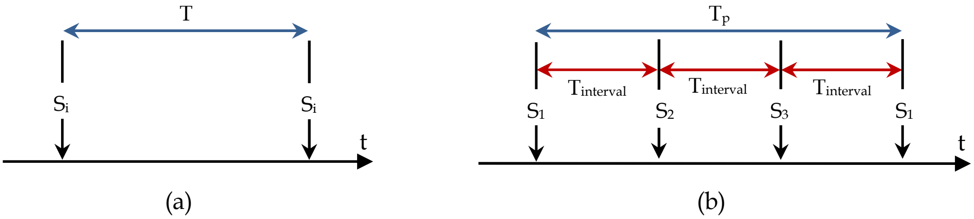


Fig. 7. (a) Period of awakening a given node in the un-cooperative scheduling. (b) Scheduling for a cluster consisting of three members (S_1, S_2, S_3).

3.3.1 Cluster-based T_P and $T_{interval}$ computation

Let us consider as baseline mechanism a non-collaborative duty-cycled scheme in which every node awakes with an interval period of time T and monitor the area (i.e., takes a picture and performs object detection) as tier 1 in (Kulkarni et al., 2005). The objective of the collaborative mechanism is to produce a cluster-based duty-cycled scheduler in which: (i) Each node is awakened and senses the area with a reliable period of T_p>T taking advantage of the overlapping among nodes in the cluster, thus, saving energy and increasing network lifetime. Each cluster will have its own T_P interval, determined according to the cluster-size and the clustering scale. (ii) During the sleeping period of each member of a given cluster, other nodes belonging to the cluster are awakened with intervals of $T_{interval} < T$ (that is equal to: T_P / C_{size}) in a sequential manner.

The area sensed by each cluster is related to the MCCD area. In order to compute T_p we will consider the MCCD area. By awaking each member of a given cluster, in average, a part of the related MCCD with a ratio equal to $1/\beta$ is captured (Equation (2)). Note that the MCCD is an area of $\beta.A_{FoV}$ and is sensed by C_{size} overlapping members, thus sensing the environment by each node delivers information not only from the FoV of the awakened node but also from some overlapped parts of the FoV of other nodes in the same cluster. Then, we may define the node interval duty-cycle period as:

$$T_P = T \cdot \frac{C_{size}}{\beta} = T \cdot \frac{C_{size}}{C_{size} - \gamma \cdot (C_{size} - 1)} \tag{3}$$

Note that the T_P is proprietary for each cluster in terms of its cluster-size and clustering scale. As it was mentioned before, the MCCD calculated by Equation (1) is the maximum covering domain of a cluster while the effective cluster covering domain may be less than MCCD since some members of a given cluster may overlap more than the region determined by γ. Consequently, a given cluster can cover an area less than $\beta \cdot A_{FoV}$. Thus, using β gives us the lowest interval T_p and thus the most reliable one since lower values of β would increase the interval T_P. On the other hand, members of a cluster are awakened sequentially to sense their environment in an intermittent way with time intervals equal to $T_{interval}$:

$$T_{interval} = \frac{T_P}{C_{size}} = \frac{T}{C_{size} - \gamma \cdot (C_{size} - 1)} \leq T \tag{4}$$

Let us consider Figure 6.b and for example a cluster with three members, $C = \{S_1, S_2, S_3\}$, cluster-head S_1 and $\gamma = 0.5$. Every node will be awakened every $T_P = 1.5 \cdot T$ seconds and the area will be monitored every $T_{interval} = 0.5 \cdot T$ seconds. As can be observed, every sensor is awakened with a period higher than the non-collaborative scheme but the area is monitored more times. Then, the area duty-cycled frequency is increased while the sensor duty-cycled frequency is reduced.

Table 2 shows the evolution respects of T_p and $T_{interval}$ to T as a function of γ for several C_{size}. We first have to notice that for a clustering scale factor $\gamma = 1$, $T_p = T$, while for $\gamma < 1$, $T \leq T_P \leq T/(1-\gamma)$. Then, the duty-cycle frequency at which a specific node is awakened is decreased by a factor that at least is $(1-\gamma)$ times the frequency of the non-collaborative scheme. On the other hand some sensor of the cluster will be on duty every $T_{interval}$ seconds. Note that $T_{interval}$ will be lower than T and will be smaller as C_{size} increases. This means that the area is monitored more frequently although every specific sensor monitors with less frequency. The reason is justified in how clusters are formed. Any sensor of the cluster overlaps with the first-member by at least an area of $\gamma \cdot A_{FoV}$. Thus, when a sensor enters in duty, he will monitor an area equal to $\gamma \cdot A_{FoV}$ overlapped with the first-member and an area equal to $(1-\gamma) \cdot A_{FoV}$ that in the worst case does not overlap with any other member of the cluster. Sensing the whole cluster area with $T_{interval}$ equal to T would result in that an area equivalent to (1−γ) ·A_{FoV} would be monitored every C_{size} ·T, a value that can be very high. However, using Equation (3), monitoring of the area equivalent to (1−γ) ·A_{FoV} is guaranteed by a monitoring interval that is not superior to T/(1−γ), that is much lower than C_{size} ·T.

γ \ C_{size}	5	4	3	2
0.5	1.67	1.60	1.5	1.33
0.55	1.79	1.70	1.58	1.38
0.6	1.92	1.82	1.67	1.43
0.65	2.08	1.95	1.77	1.48
0.7	2.27	2.11	1.88	1.54

(a)

γ \ C_{size}	5	4	3	2
0.5	0.334	0.4	0.5	0.665
0.55	0.358	0.425	0.527	0.690
0.6	0.384	0.455	0.557	0.715
0.65	0.416	0.488	0.590	0.740
0.7	0.454	0.528	0.627	0.770

(b)

Table 2. (a) T_p/T , (b) $T_{interval}$/T for different cluster sizes and clustering scales.

Sleep/wake up protocols has extensively been studied in the area of wireless sensor networks, mainly for the radio subsystem, (Anastasi et al., 2009). Our clustering algorithm works on the sensing subsystem. It is important to notice that executing object detection does not imply sending packets to the sink. Thus, the sleep/wake up algorithm can be decoupled with the radio subsystem. Sleep/wake up can be based on periodic duty-cycle synchronized by the first-member: every T_p period, the sensing subsystem wakes up and performs object detection. However, clock drifts can cause cluster de-synchronization. To handle resynchronization, the system makes use of the beaconing scheme for cluster maintenance: nodes receive periodical beacons from the first-member and vice versa in order to detect new members or to detect members that have died. Beaconing duty-cycling belongs to the radio subsystem and it is independent of the sensing subsystem. That means that waking up the sensor to send a beacon is independent of waking up the sensor to take a picture and perform object detection. Thus, the cluster-head may resynchronize cluster members without need of waking up the sensing subsystem.

3.4 Lifetime prolongation evaluation

To evaluate the scheduling scheme in terms of power conservation, we compare the cooperative scheduled scheme with a single-tier network or a tier of a multi-tier architecture consisting of N nodes monitoring without coordination among them as (Rahimi et al., 2005; Kulkarni et al., 2005; Feng et al., 2005), in which, nodes are awakened with a time period of T. We note that the evaluation is over the sensing subsystem and that the radio subsystem (*i.e.;* transmission and reception of packets) is not taken into account.

The energy consumed in the network for object detection by N nodes during a duty-cycle interval of T in the non-collaborative scheduling is:

$$E = N \cdot (T_{sleep} \cdot P_{sleep} + E_{w\_up} + E_{cap} + E_{detect}) \tag{5}$$

where T_{sleep} and P_{sleep} are the period and power consumption for a node in sleep mode. E_{w_up}, E_{cap} and E_{detect} respectively are the energies consumed in waking up a node, capturing a picture and performing object detection.

Let us now consider the cooperative scheduling algorithm in a clustered tier/network. Both, the interval between waking up consecutive nodes in the same cluster and the period of waking up a given node are functions of the cluster-size of the cluster which the nodes belong to. In one hand, in clusters with high cluster-size, $T_{interval}$ is small and thus cluster duty-cycle frequency is increased. On the other hand, higher number of nodes in the cluster causes longer periods T_P for awaking a given node of the cluster and thus yields an enhancement for power conservation in cluster's members. Assuming average cluster-size for all clusters in the tier/network, T_P will be:

$$T_P = \frac{T \cdot \mu_{C_{size}}}{\mu_{C_{size}} - \gamma \cdot (\mu_{C_{size}} - 1)} \tag{6}$$

where T is the base period for waking nodes in the base un-coordinated tier. Figure 8 shows the evolution of T_p normalized by T (*i.e.*; μ_{Csize}/β) for several node densities and clustering scales, γ. We may observe that the node average duty-cycle frequency is reduced by factors that are, for example, on the order of 0.78 for a 200 node network and a scale factor of γ = 0.6.

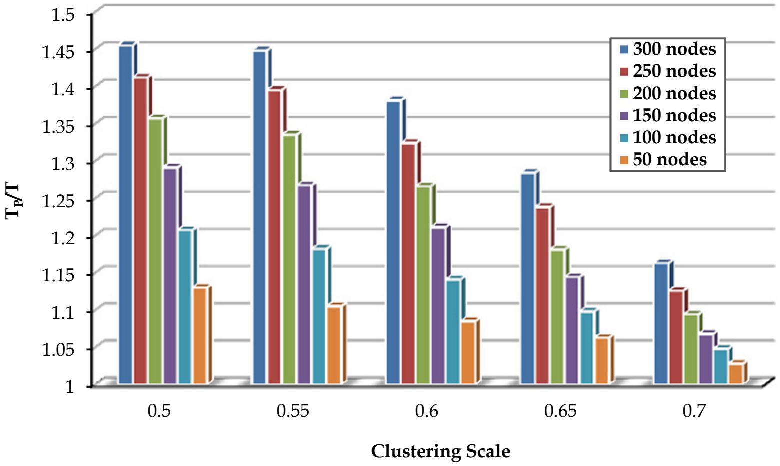


Fig. 8. T_P/T for several node densities and clustering scales.

Consequently, the total amount of averaged consumed energy by nodes for object detection in the coordinated tier during T_P will be:

$$E_P = E + N \cdot P_{sleep} \cdot (T_P - T) \tag{7}$$

From (6) and (7):

$$E_P = E + \frac{\gamma \cdot T \cdot (\mu_{C_{size}} - 1)}{\mu_{C_{size}} - \gamma \cdot (\mu_{C_{size}} - 1)} \cdot N \cdot P_{sleep} \tag{8}$$

So:

$$\frac{E_P}{T_P} = \frac{E \cdot (\mu_{C_{size}} - \gamma \cdot (\mu_{C_{size}} - 1))}{T \cdot \mu_{C_{size}}} + \frac{\gamma \cdot (\mu_{C_{size}} - 1) \cdot N \cdot P_{sleep}}{\mu_{C_{size}}}$$

$$\frac{E_P}{T_P} = (1 - \frac{\mu_{C_{size}} - 1}{\mu_{C_{size}}} \cdot \gamma) \cdot \frac{E}{T} + \frac{N \cdot \gamma \cdot (\mu_{C_{size}} - 1)}{\mu_{C_{size}}} \cdot P_{sleep} \quad \textit{where } (0 < \gamma < 1) \textit{ and } (\mu_{C_{size}} > 1)$$

Therefore, the consumed power is:

$$P_P = \lambda \cdot P + \sigma \cdot P_{sleep} \tag{9}$$

where:

$$\lambda = (1 - \frac{\mu_{C_{size}} - 1}{\mu_{C_{size}}} \cdot \gamma) \quad , \quad 0 < \lambda < 1$$

$$\sigma = \frac{N \cdot \gamma \cdot (\mu_{C_{size}} - 1)}{\mu_{C_{size}}} \quad , \quad 0 < \sigma < \gamma \cdot N$$

Parameter P in Equation (9) is the power consumed in the network with the base un-coordinated mechanism. The consumed power in our scheme (P_P) is reduced by a factor λ with respect to P. The λ factor depends on the average cluster-size and the clustering scale factor. As can be observed from Equation (9) increasing μ_{Csize} produces lower values of λ, and thus a saving in energy with respect the uncoordinated system. For example a μ_{Csize} =1.5 (100 nodes with γ=0.5) produces a λ = 1 – γ/3 = 0.83 while a μ_{Csize} = 2.15 (200 nodes with γ = 0.5) produces a λ = 1–0.53 γ = 0.73. The other term (σ·P_{sleep}) in Equation (9) is due to the fact of taking nodes to sleep mode in intervals of duration (T_P > T) and then nodes sleep T_p–T more time than in the un-clustered scheme.

Figure 9 illustrates the impact of factor λ in Equation (9) in terms of node densities for several clustering scales. From this figure we can see that in high node density tiers, the factor λ is more beneficial since μ_{Csize} is higher and thus there is more potential of cooperation among nodes.

Figure 10 shows the consumed power (P) in the base un-coordinated tier for object detection in four cases of period of duty-cycle for different node densities. The consumed power has been computed for nodes consisting of Cyclops as camera sensor embedded in the host MICA II, similar to the tier 1 in (Kulkarni et al., 2005).

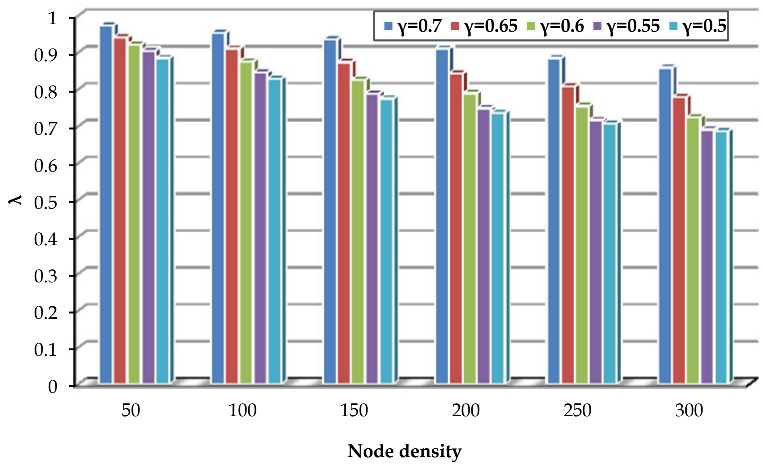


Fig. 9. Factor λ in cooperative scheduling for several clustering scales.

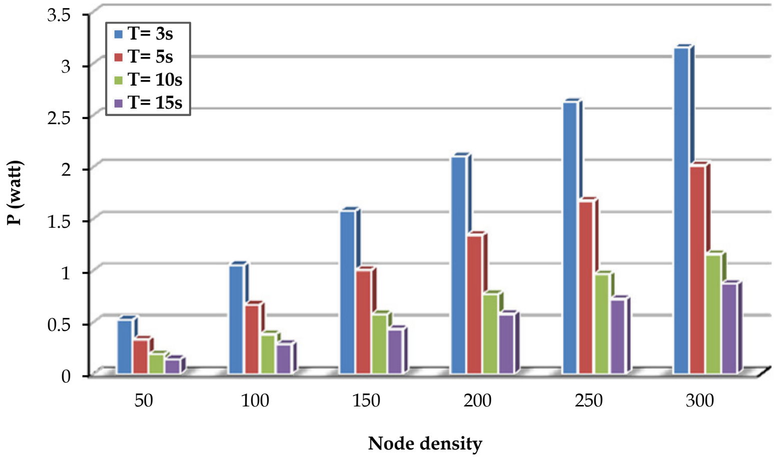
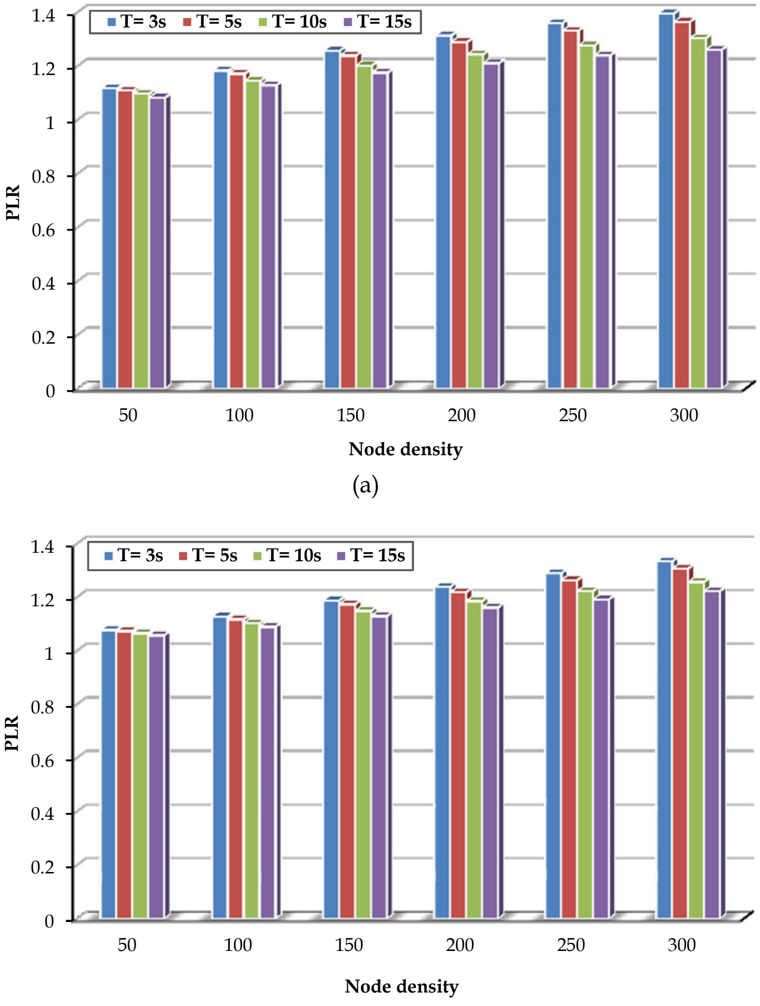


Fig. 10. Consumed power (P) for a non-cooperative tier/network of nodes consisting of Cyclops.

For instance, in the case without coordination, the power consumed in a tier consisting of 200 nodes that performs monitoring with a duty cycle of T=5 second, is 1.344 watts. In the coordinated network with the same number of nodes and a clustering scale of 0.5, the power consumed by the network would be reduced by a factor λ of 0.737 (see Figure 9) at the cost of increasing 52.60 mW, ($\sigma \cdot P_{sleep}$). This means a tier power consumption of $1.344 \cdot 0.737 + 0.0526 = 1.043$ Watts implying a reduction of 22.39%. Thus, in this case, the Prolongation Lifetime Ratio (PLR) would be of 1.344/1.043 = 1.289. Figure 11.a,b shows the prolongation lifetime ratio assuming a clustering scale of 0.5 and 0.6 for different node densities in four cases of duty-cycle (T). Tiers with high number of nodes have higher capability for cooperation and thus their nodes can conserve considerable amount of energy comparing to sparse networks and consequently, have longer prolonged lifetime. The figure indicates the more prolongation lifetime for dense tiers.



(a)

(b)

Fig. 11. Prolongation Lifetime Ratio (PLR) for different node densities in the clustered tier with a clustering scale equal to (a) 0.5. (b) 0.6, in four states of base awakening period.

4. Future work

In the clusters established by the depicted mechanism, each cluster member has a common sensing region with the CH. The clusters do not have any intersection and each cluster monitors its covering domain with only intra-cluster collaboration. Clustering with the capability of intersection and cooperation among clusters can increase the scale of efficiency of monitoring performance and power conservation of cluster members. In a monitoring mechanism utilizing intra and inter cluster cooperation, sensing regions are allocated to intersected clusters thus can be monitored with a higher frequency and/or consuming less amount of energy although the node selection and scheduling procedure will be more complicated. Some initial work has been done in (Alaei & Barcelo, 2010).

5. Conclusion

In this chapter a mechanism for management the wireless multimedia sensor nodes, was described. The mechanism, first, clusters nodes according to their scale of similarity in covering the environment; second, selects and schedules members of established clusters to monitor the sensing region which is divided among clusters. The members of each cluster are scheduled with an exclusive frequency based on the number of members in the cluster and the scale of overlapping among fields of view of the cluster members and thus the monitoring efficiency is increased. Moreover, because of the established intra cluster coordination and collaboration, sensing subsystem of multimedia nodes are optimized to avoid redundant and overlapped sensing. Thus, the capability of energy saving is considerably enhanced with respect to ordinary duty-cycling manners of environment monitoring by WMSNs. On the other hand, optimizing the data sensed by sensing subsystem results in conservation of energy in the transmission and processing subsystems since they meet less amounts of multimedia data to be transmitted and/or processed by the network nodes. Results show how this mechanism prolongs the network lifetime along with a better monitoring performance.

6. Acknowledgment

This work is partially supported by the EuroNF NoE and grants TIN2010-21378-C02-01 and SGR2009-1167.

7. References

Adriaens, J.; Megerian, S. & Potkonjak, M. (2006). Optimal worst-case coverage of directional field-of-view sensor networks, *Proceedings of the 3rd IEEE Communication Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (IEEE SECON)*, pp. 336–345, ISBN 1-4244-0626-9, Reston, VA USA, September 25–28, 2006

Akyildiz, I. F.; Melodia, T. & Chowdhury, K. R. (2007). A Survey on Wireless Multimedia Sensor Networks. *Computer Networks*, Vol. 51, Issue 4, (March 2007), pp. 921-960, ISSN 1389-1286

Alaei, M. & Barcelo-Ordinas, J.M. (2010). A method for clustering and cooperation in Wireless Multimedia Sensor Networks. *Sensors*, Vol. 10, No. 4, (March 2010), pp. 3145-3169, ISSN 1424-8220

Alaei, M. & Barcelo-Ordinas, J.M. (2010). MCM: multi-cluster-membership approach for FoV-based cluster formation in wireless multimedia sensor networks, *Proceedings of The 6th International Wireless Communications and Mobile Computing Conference (IWCMC 2010)*, pp. 1161-1165, ISBN 978-1-4503-0062-9, Caen, France, June 28-July 2, 2010

Alippi, C.; Anastasi, G.; Galperti, C.; Mancini, F. & Roveri, M. (2007). Adaptive Sampling for Energy Conservation in Wireless Sensor Networks for Snow Monitoring Applications, *Proceedings of IEEE International Workshop on Mobile Ad-hoc and Sensor Systems for Global and Homeland Security (MASS-GHS07)*, Pisa, Italy, October 8, 2007

Anastasi, G.; Conti, M.; Francesco, M. & Passarella, A. (2009). Energy conservation in wireless sensor networks: A survey. *Ad Hoc Networks*, Vol. 7, Issue 3, (May 2009), pp. 537–568, ISSN 1570-8705

Chen, P.; Ahammed, P.; Boyer, C.; Huang, S.; Lin L.; Lobaton, E.; Meingast, M.; Oh, S.; Wang, S.; Yan, P.; Yang, A.Y.; Yeo, C.; Chang, L.C.; Tygar, D. & Sastry, S.S. (2008). CITRIC: A low-bandwidth wireless camera network platform. *Proceedings of the 2nd ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC 2008)*, pp. 1–10, ISBN 978-1-4244-2665-2, Palo Alto, CA, USA, September 7–11, 2008

Dagher, J. C.; Marcellin, M. W. & Neifeld, M. A. (2006). A method for coordinating the distributed transmission of imagery, *IEEE Transactions on Image Processing*, Vol. 15, No. 7, (July 2006), pp. 1705–1717, ISSN 1057-7149

Diamond, D. (2006). Energy Consumption Issues in Chemo/Biosensing using WSNs, *Energy and Materials: Critical Issues for Wireless Sensor Networks Workshop*, June 30, 2006.

Ercan, A.; Gamal, A. E. & Guibas, L. (2006). Camera network node selection for target localization in the presence of occlusions, *Proceedings of the ACM SenSys Workshop on Distributed Smart Cameras*, 2006.

Feng, W.C.; Kaiser, E.; Shea, M.; Feng, W.C & Baillif, L. (2005). Panoptes: scalable low-power video sensor networking technologies. *ACM Transactions on Multimedia Computing, Communications, and Applications*, Vol. 1, Issue 2, (May 2005), pp. 151–167, ISSN 1551-6857

Kerhet, A.; Magno, M.; Leonardi, F.; Boni, A. & Benini, L. (2007). A low-power wireless video sensor node for distributed object detection. *Journal of Real-Time Image Processing*, Vol. 2, No. 4, October 2007, pp. 331–342, ISSN 1861-8219

Kulkarni, P.; Ganesan, D.; Shenoy, P. & Lu, Q. (2005). SensEye: A multi tier camera sensor network, *Proceedings of the 13th ACM International Conference on Multimedia (ACM MM 2005)*, pp. 229–238, ISBN 1-59593-044-2, Singapore, November 6–11, 2005

Margi, C.B.; Lu, X.; Zhang, G.; Stanek, G.; Manduchi, R. & Obraczka, K. (2006). Meerkats: A power-aware, self-managing wireless camera network for wide area monitoring, *Proceedings of International Workshop on Distributed Smart Cameras (DSC 06) in conjunction with SenSys06*, ISBN 1-59593-343-3, Boulder, CO, USA, October 31, 2006

Pahalawatta, P. V.; Pappas, T. N. & Katsaggelos, A. K. (2004). Optimal sensor selection for video-based target tracking in a wireless sensor network, *Proceedings of the International Conference on Image Processing (ICIP '04)*, pp. 3073– 3076, ISBN 0-7803-8554-3, Singapore, October 24-27, 2004

Park, J.; Bhat, P. & Kak, A. (2006). A look-up table based approach for solving the camera selection problem in large camera networks, *Proceedings of the International Workshop on Distributed Smart Cameras (DCS '06) in conjunction with SenSys06,* ISBN 1-59593-343-3, Boulder, CO, USA, October 31, 2006

Pereira, F.; Torres, L.; Guillemot, C.; Ebrahimi, T.; Leonardi, R. & Klomp, S. (2008). Distributed video coding: Selecting the most promising application scenarios. *Signal Processing: Image Communication*, Vol 23, Issue 5, (June 2008), pp. 339–352, ISSN 0923-5965

Pottie, G. & Kaiser, W. (2000). Wireless Integrated Network Sensors. *Communication of the ACM*, Vol. 43, N. 5, (May 2000), pp. 51-58

Raghunathan, V.; Ganeriwal, S. & Srivastava, M. (2006). Emerging techniques for long lived wireless sensor networks. *IEEE Communications Magazine*, Vol. 44, Issue 4, (April 2006), pp. 108- 114, ISSN 0163-6804

Raghunathan, V.; Schurghers, C.; Park, S. & Srivastava, M. (2002). Energy-aware Wireless Microsensor Networks. *IEEE Signal Processing Magazine*, Vol. 19, Issue 2, (March 2002), pp. 40-50, ISSN 1053-5888

Rahimi, M.; Baer, R.; Iroezi, O.I.; Garcia, J.C.; Warrior, J.; Estrin, D. & Srivastava, M. (2005). Cyclops: in situ image sensing and interpretation in wireless sensor networks, *Proceeding of the 3rd ACM Conference on Embedded Networked Sensor Systems (SenSys 05)*, pp.192–204, ISBN 1-59593-054-X ,San Diego, CA, USA, November 2–4, 2005

Rowe, A.; Rosenberg, C. & Nourbakhsh. I. (2002). A Low Cost Embedded Color Vision System. *Proceedings of the international IEEE/RSJ Conference on Intelligent Robots and Systems (IROS 2002)*, pp. 208-213, ISBN 0-7803-7398-7, Lausanne, Switzerland, Sep.30-Oct.4, 2002

Schott, B.; Bajura, M.; Czarnaski, J.; Flidr, J.; Tho, T. & Wang, L. (2005). A modular power-aware microsensor with >1000X dynamic power range, *Proceedings of the Fourth International Symposium on Information Processing in Sensor Networks (IPSN 2005)*, pp.469-474, ISBN 0-7803-9201-9, UCLA, Los Angeles, California, USA ,April 25-27 2005

Soro, S. & Heinzelman, W. (2007). Camera selection in visual sensor networks, *Proceedings of the IEEE Conference on Advanced Video and Signal Based Surveillance (AVSS '07)*, pp. 81–86, ISBN 978-1-4244-1696-7, London, UK, September 5- 7, 2007.

Soro, S. & Heinzelman, W. (2005). On the coverage problem in video-based wireless sensor networks. *Proceedings of the 2nd IEEE International Conference on Broadband Communications and Systems (BroadNets)*, pp. 932–939, ISBN 0-7803-9276-0, Boston, MA, USA, October 3–7, 2005

Tavli, B.; Bicakci, K.; Zilan, R. & Barcelo-Ordinas, J.M. (2011). A Survey of Visual Sensor Platforms. *Journal on Multimedia Tools and Applications*, ISSN 1573-7721, June 2011

Tezcan, N. & Wang, W. (2008). Self-orienting wireless multimedia sensor networks for occlusion-free viewpoints. *Computer Networks*, vol. 52, issue 13, (September 2008), pp. 2558–2567, ISSN 1389-1286

Zamora, N. H. & Marculescu, R. (2007). Coordinated distributed power management with video sensor networks: analysis, simulation, and prototyping, *Proceedings of the 1st ACM/IEEE International Conference on Distributed Smart Cameras (ICDSC '07)*, pp. 4–11, ISBN 978-1-4244-1354-6, Vienna, Austria, September 26-28, 2007.

http://www.ti.com/product/tmp103?DCMP=analog_signalchain_mr&HQS=Other%252bPR%252btmp103-pr