Maximizing Angle Coverage in Visual Sensor Networks

Kit-Yee Chow, King-Shan Lui and Edmund Y. Lam
Department of Electrical and Electronic Engineering
The University of Hong Kong
Pokfulam Road, Hong Kong, China

Abstract— **In this paper, we study the angle coverage problem in visual sensor networks where all sensors are equipped with cameras. An object of interest moves around the network and the sensors near the object are responsible for capturing images of it. The angle coverage problem aims to identify a set of sensors that preserve all the angles of view of the object while fulfilling the image resolution requirement. The user is required to specify the minimum acceptable image resolution in the request. Only the images that fulfill the resolution requirement will be considered. In order to save transmission energy, the number of images to be sent should be minimized. We develop a distributed algorithm to identify the minimum set of sensors such that all these images cover the maximum angle of view of the target. Our simulation results show that our protocol can achieve significant reduction in transmission load while preserving the widest angle of view.**

I. Introduction

Wireless sensor networks have become an emerging technology for surveillance, disaster management, habitat monitoring, etc [1], [2], [3]. A sensor is a very small device and it is powered by battery which is unlikely to be rechargeable. This limitation in energy puts extra constraints in the operation of a sensor. Message transmission has been shown to be the major source of energy dissipation in sensor networks [1]. In order to prolong the system lifetime, we should keep the transmission load as small as possible.

One fundamental issue in sensor networks is the coverage problem, which reflects how well a target or an area is monitored or tracked by sensors. In traditional target or area coverage problem, a set of sensors is identified such that each given region or target is covered by at least k sensors in the set [4]. In this paper, we consider the problem of *angle coverage* instead of area/target coverage in sensor networks.

We consider a tracking system that the object of interest moves around the network and the sensors around it are responsible for capturing the images of it. All sensors are equipped with cameras and they are randomly distributed in the network with arbitrary orientations. The sensors near the object of interest will capture the images of the moving object. Due to the different orientations and different distances from the object of the cameras, the images captured are of different resolution and cover different sides of the object. A trivial way to preserve the widest angles of view is to gather all the images. Very often, because sensor networks are dense, the entire $360°$ view of the object is captured. However, this is not an energy efficient approach as there may be many redundant images and this heavy transmission load will drain the batteries of sensors quickly. Additionally, users may want different image resolution based on different applications and targets. Given a certain image resolution requirement, we would like to identify a minimum set of sensors such that the images captured by these sensors can fulfill the resolution requirement and they can preserve all the angles of view.

We develop an efficient distributed algorithm to identify the minimum cover set. The algorithm has been tested through extensive simulations and we formally prove the correctness of it.

The rest of the paper is organized as follows: Section II presents the related work. Section III describes the network model. Section IV describes the protocol, and simulation results are shown in Section V. We finally conclude our paper in Section VI.

II. Related Work

Coverage problem is an important issue in sensor networks. There are many research activities studying this problem. In [5], the authors study the problem of detecting and eliminating redundancy in a sensor network while preserving the network's coverage and a distributed algorithm has been developed to solve the problem. Wang et al. [6] proposed a protocol that dynamically configures the network in order to provide different coverage degrees requested by the application while maintaining connectivity. Gupta et al. [7] have designed techniques to exploit the redundancy in the sensor network by selecting a small subset of sensor, called connected sensor cover, that is sufficient to process a given query.

Ref [8] also studies the coverage problem but specifically in video-based sensor networks. Unlike traditional sensors, video cameras have the unique feature of capturing images of objects that are not necessarily in the camera's vicinity. The objects covered by the camera can be distant from the camera. In video-based sensor networks, the sensing range of sensor nodes is replaced with camera's field of view. In this work, the authors considered the situation when all the camera nodes are mounted on a plane and they are directed towards the service plane. They showed that because of the unique way that camera capture data, the algorithm does not give expected results in terms of coverage preservation of monitored areas.

All of works mentioned above consider the *area coverage* problem in the sensor networks, they aim at preserving the

1-4244-0353-7/07/$25.00 ©2007 IEEE

coverage of the monitored areas. In a visual sensor network, it is not common that all the cameras monitor the area in one plane as the case in [8]. Usually, the sensors are distributed randomly in the network with arbitrary orientations. In this paper, we consider the problem of *angle coverage* instead of area coverage in sensor networks. In a tracking system, the object of interest moves around the network and the sensors around it are responsible to capture the images of it. The nodes may capture the object from different directions. Suppose the object of interest is an elephant, we would like to have a round view of it, not only observing the head or the tail. We aim at preserving all the angles of view captured by the sensors. We study a variable of this problem in [9].

III. Network Model

We consider a sensor network where all sensor nodes are camera-equipped and are identical in terms of intrinsic parameters such as field-of-view (FOV). Due to size limitation, the cameras do not have the capability of pan, tilt and zoom and they are static. All sensor nodes are randomly distributed with arbitrary orientations. Each camera node is able to know its physical location by means of GPS [10] or a localization algorithm [11]. Additionally, image-based localization algorithms such as [12], [13] enable a sensor node to know its physical location and camera orientation as well. The sensors that lie within the transmission range of sensor i would be the neighbors of i, they can communicate to each other.

A. Image Capture

The object of interest moves around the network and the main task of the sensor is to track the object of interest by taking images of it. We assume that all camera nodes possess the capability of object recognition [14], [15], they can recognize whether the object of interest is on the images or not. The images captured are kept in the local memory of the sensors and they will be sent to the sink when necessary.

B. Relationship Between Image Resolution And Node Distance

For simplicity, the object of interest is approximated with a cylindrical shape with radius R_o as illustrated in Figure 1. Let 2α be the FOV of the camera node, 2β be the captured angle and R_1 be the distance between the camera node and the center of the object of interest.

Suppose the raw image size of the camera node is 512×512. As the case in Figure 1(a), the full view of the captured image is occupied by the object of interest and $2y$ meters long visual data are projected onto 512 pixels horizontally. We may say that the image resolution is $\frac{512}{2y}$ pixels per meter. As the camera node is farther away from the target, x increases and thus y increases. As y increases, the image contains more visual data of the object of interest. In other words, for the same amount of visual data, they are represented by fewer number of pixels as the camera node is farther away and hence the resolution decreases. This shows that image resolution and node distance are inversely proportional to each other.

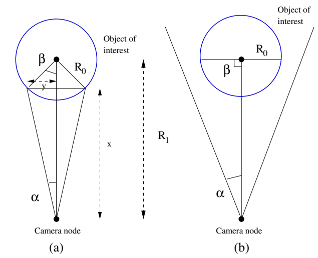

Fig. 1. (a) Small FOV, (b) large FOV

In Figure 1, 2β is the captured angle of the cylindrical object and can be evaluated by the following formula:

$$\tan(\beta) = \frac{y}{R_1 - x} = \frac{y}{\sqrt{{R_o}^2 - y^2}} \quad (1)$$

According to Equation 1, β increases as y increases. Since y is directly proportional to node distance and FOV, we can deduce that under the same FOV, the longer the node distance, the larger the captured range. Similarly, under the same node distance, the larger the FOV, the larger the captured range. As illustrated in Figure 1, the largest captured angle would be $180°$. If this is the case, 2 images would be enough to give a round view of the target. However, it is not desirable as the side view of the object of interest cannot be seen clearly.

Intuitively, larger β would facilitate the system to require fewer images to cover $360°$ and less energy would be needed. It implies that we should select images which are taken farther away from the object or by a camera with larger FOV. However, the resolution will be reduced if the node distance is increased. Consequently, there is a tradeoff between resolution and number of images needed.

C. Angle Coverage

In the tracking system, all the sensors around the target will capture images of it from different directions. Ideally, we would like to view the target in all directions, i.e. $360°$ angle coverage. A trivial way to preserve all the angles of view is to gather all the images. However, in visual sensor networks, the camera nodes are close to each other and thus the images captured are highly correlated. Sending all the images to the sink will be a waste of energy.

Referring to the network in Figure 2 where the object of interest is represented by a shaded circle, the white nodes {A, B, C, ..., O, P} are the sensors that will capture images of the object. An image is said to be redundant if its removal does not affect the angle coverage. Both {M, D, I, O} and {D, I, O} are able to give a $360°$ coverage, the image of node M is redundant as the elimination of it does not affect the angle coverage. In order to reduce the transmission load, redundant images should be eliminated.

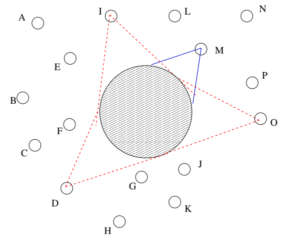

Fig. 2. Network Model

D. User Request

When a user issues a request, the request will be delivered to the sensors that are around the object of interest by a routing protocol, such as GPSR [16]. Upon receiving the request, each sensor determines on its own whether it should send its image back to the user. A minimum acceptable resolution, κ, is specified by the user in terms of number of pixels per meter in the request. Based on this value, our protocol will find out the minimum number of images which fulfill the resolution requirement while preserving $360°$ angle coverage. κ can be a default value known to all sensors or, it can be specified by the user on an object-by-object basis.

IV. PROTOCOL

In this section, we describe how a sensor identifies whether its image is wanted. In the environmental monitoring system, the object of interest can be an elephant, a tiger or a panda. Although it is difficult for us to know the exact size of the object of interest, we can approximate the target's size based on statistics.

There are two phases in our protocol: Candidate Identification and Image Selection. The first phase prunes images of low resolution while the second phase identifies images that cover all angles of view.

A. Candidate Identification

As illustrated in Section III-B, given the node distance (R_1), FOV (2α), and object radius (R_o), we can approximate the image resolution by $\frac{512}{2y}$ pixels per meter if the image size is 512×512. All the sensor nodes are capable of calculating its approximate image resolution. Upon receiving a request, the sensor can determine whether it is a candidate or not by comparing the resolution requirement κ with its own approximated image resolution. If the approximated resolution is higher or equal to κ, the sensor can proceed to the Image Selection phase.

B. Image Selection

After the Candidate Identification process, all the images with resolution lower than the requested value are eliminated. However, there are still many redundant images. In order to minimize the transmission load, only the sensors belong to the minimum subset of sensors which preserve the angle coverage will send their images to back to the user.

As illustrated in Section III-B, the angle of view covered by a node (2β) can be calculated using R_1, R_o and 2α. Since every node knows its orientation, a node can specify the angle of view as $[\sigma, \sigma+2\beta]$. For example, we can set East as $0°$ and an image that spans from East to Southeast would be $[0°, 45°]$. When $[\sigma, \sigma+2\beta]$ is found, each node should broadcast this information to its neighbors.

Before we get into the details of how a node determines whether its image is requested based on the information collected from neighbors, we first formally define the problem and present a centralized mechanism.

1) Problem Definition: Given a set of sensors S, let the angle of view covered by sensor node $i \in S$ be $[s_i, t_i]$. If $s_i > t_i$, it means the image spans across $0°$. A set $V \subseteq S$ is a **cover** if for each angle $\gamma \in [0°, 360°)$, there exists a sensor i in V such that $\gamma \in [s_i, t_i]$. A cover C is a **minimum cover** if it is smallest in size among all covers. That is, $|C| \leq |V|$ for every cover V.

Refer to the example in Figure 3, both {1, 4, 7} and {2, 3, 5, 6} are covers while {1, 4, 7} is also a minimum cover. Minimum cover is not necessarily unique. In the example, {1, 3, 7} is also a minimum cover. Due to the randomness of sensor distribution, it is possible that there are not enough sensors to cover $360°$. We aim at finding the set of sensors with smallest size which preserves all the captured range. Refer to the example in Figure 3, if sensors 6 and 7 are removed, {1, 4, 5} or {1, 3, 5} or {2, 3, 5} are a minimum set of sensors that retain all the captured ranges.

To tackle the problem, we first explain the solution in the case where there is no image spanning across $0°$. Intuitively, to reduce the number of ranges needed, we should try to select the range that covers as much view that has yet been covered as possible. Suppose that the images selected cover $[0°, current\_angle]$. The next range to be included should start before $current\_angle$ and end as far as possible. Then, the view that is remained uncovered will be minimized. The pseudocode of the algorithm is as follows:

FIND_MIN_COVER(C, S)

$C \leftarrow \emptyset$
find j where $t_j \geq t_i \forall i \in S$
$max\_angle \leftarrow t_j$
$current\_angle \leftarrow 0$
while ($current\_angle < 360$)
{
  $D = \{i \mid current\_angle \in [s_i, t_i)\}$
  if ($D = \emptyset$) {
    if ($current\_angle == max\_angle$) {
      Return
    }
    else {
    find k where
    $0 < s_k - current\_angle \leq s_i - current\_angle$
    $\forall i \in S - C$
    $current\_angle \leftarrow s_k$
    }
  }
  else {
    find j where $t_j \geq t_i \forall i \in D$
    $C \leftarrow C \cup \{j\}$
    $current\_angle \leftarrow t_j$
  }
}

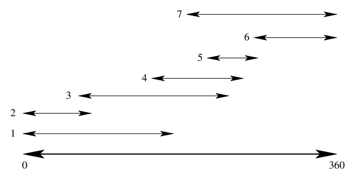

Fig. 3. Cover Example

Lemma 1: Algorithm FIND_MIN_COVER is correct.

Proof: To prove the lemma, we have to argue that Algorithm FIND_MIN_COVER finds a minimum cover if it exists or finds a minimum set of sensors that preserves all the angles of view.

Suppose that for a certain S, there exists a minimum cover or minimum set C' of size k'. The angles of view of those sensors are $[\sigma_1, \tau_1], [\sigma_2, \tau_2], ..., [\sigma_{k'}, \tau_{k'}]$, where $\tau_i < \tau_{i+1}$ for $1 \le i < k'$. We further assume C returned by the algorithm is of size $k \ge k'$ and the angles of view of those in C are $[s_1, t_1], [s_2, t_2], ..., [s_k, t_k]$, where $t_i < t_{i+1}$ for $1 \le i < k$. Note that $\sigma_1 = s_1 = 0$ and $\tau_{k'} = t_k = max\_angle$. From Line 2, max_angle is the maximum ending angle among all the sensors. If C' is a minimum cover, $max\_angle = 360°$. If we can show that $\tau_i \le t_i$ for $1 \le i \le k'$, we prove $k = k'$ and complete the proof.

According to the algorithm, among those angles of view that start from 0, t_1 should be the largest ending angle. That is, $\tau_1 \le t_1$. By Mathematical Induction, we can show that $\tau_i \le t_i$ for $1 \le i \le k'$ as follows:

If C' is a minimum cover, $\sigma_i \le \tau_{i-1}\ \forall i$. That is, we can assume $\sigma_i \le \tau_{i-1} \le t_{i-1}$. There are two cases:

- Case 1: $\tau_i \le t_{i-1}$
  It is obvious that $\tau_i \le t_i$.
- Case 2: $\tau_i > t_{i-1}$
  After setting $current\_angle$ to t_{i-1} in Line 20 of the algorithm, sensor of range $[\sigma_i, \tau_i]$ is a candidate in D since the range covers t_{i-1}. Among all the candidates, the largest ending angle is selected to be included in C. Hence, $\tau_i \le t_i$.

If C' cannot cover 360°, there is a gap between the capture ranges. We assume that $\tau_{i-1} \le t_{i-1}$. There are two cases:

- Case 1: $\sigma_i > \tau_{i-1}$
  As illustrated in Figure 4, there is a gap between the capture ranges. It is impossible that $t_{i-1} > \tau_{i-1}$ and thus $\tau_{i-1} = t_{i-1}$. And s_i is the nearest angle that should be included, $\sigma_i = s_i$. Since the largest ending angle is selected to be included in C, $\tau_i \le t_i$.

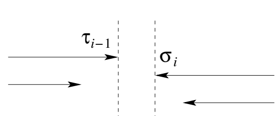

Fig. 4. Gap between capture ranges

- Case 2: $\sigma_i \le \tau_{i-1}$
  Same as the proof when C' is a minimum cover.

□

Algorithm FIND_MIN_COVER works when we can define a starting $current\_angle$ in Line 4. It may not be appropriate for us to set the $current\_angle$ to 0 if there is an image spanning across 0°. For example, in Figure 5, the capture ranges of sensors 1 and 2 span across 0°. If $current\_angle$ is set to be 0, Node 2 is the starting node. And Node 3 will be the next sensor. Eventually, a cover {2, 3, 5, 7, 8, 1} will be identified but this is not a minimum cover since Node 2 is redundant.

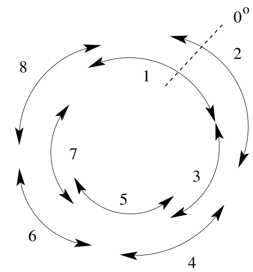

Fig. 5. Example of image spanning across 0°

To solve the problem, we first determine whether there is any angle of view that is covered by one sensor, say i, only. If this is the case, C must contain i and the $current\_angle$ in Line 4 is set to be t_i. We only have to find a minimum cover starting from angle t_i and wraps around to angle s_i. Referring to the example in Figure 5, Nodes 1, 3, 5, 7 and 8 must be selected. Suppose Node 1 is chose to be the starting node, $current\_angle$ is set to be t_1. According to the pseudocode, Node 3 will be the next sensor. Eventually, the minimum cover {1, 3, 5, 7, 8} will be identified.

If every angle is covered by more than one sensor, we identify the sensors that cover a certain angle, say 0°. For each sensor i that covers 0^o, the minimum cover from $[t_i, s_i]$ is found. The smallest minimum cover is then selected.

2) Distributed Algorithm: Algorithm FIND_MIN_COVER is a centralized algorithm and is not suitable in sensor networks since sensors only have limited memory and cannot keep global information. It is necessary for us to develop a distributed version of FIND_MIN_COVER. In the distributed algorithm, each sensor needs to obtain the angles of view of their neighbors only. That is, if the angle of view covered by sensor i is $[s_i, t_i]$, i knows the angle of view of neighbor j if $s_i \le s_j \le t_i$ or $s_i \le t_j \le t_i$. Referring to the example in Figure 3, Sensor 1 knows the angles of view of neighbors 2, 3, and 4. Referring to the example in Figure 2, you may wonder if I and O are neighbors as there are overlappings in their captured ranges. However, this would not be the case. Since there are limitations in camera sensing range and resolution, only the nodes near to the object of interest will be considered and their captured ranges will be small. The capture ranges of I and O are nearly 120°, this is so large that the image resolution would be very low and thus these two nodes will be eliminated in the Candidate Identification process.

We first describe how a node knows whether it is in a minimum cover. As mentioned earlier, sensor node k must be in a minimum cover if it covers an angle that is not covered by others. We refer such a node as **default member**. After getting the angles of view from neighbors, k determines whether it is a default member. If so, it announces to its neighbor that it is selected and then starts to identify the next sensor. According to the algorithm, if a sensor k is selected, $current\_angle$ will be set to t_k. The next sensor to be included in C must be a sensor in D in Line 7. D is the set of neighbors j of k where $s_j \leq t_k \leq t_j$. k can identify D with its local neighborhood information. Therefore, k can identify the next sensor, say i, to be included in C. k then informs i and i can find the next sensor accordingly.

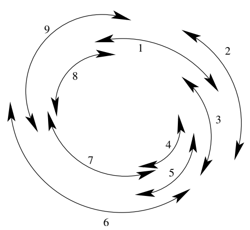

Fig. 6. Illustration of Distributed Version

C. Example

We now illustrate the whole process using the example in Figure 6. After getting neighborhood information, each node checks whether it is a default member. In this example, Node 1 is a default member. Then, Node 1 informs its neighbors, 2, 3, 8, and 9, that it is selected. It also finds the next sensor to be included in the minimum cover. It should be noted that the searching procedure can be done in either clockwise or anticlockwise direction, but not both. Suppose the clockwise direction is adopted in this example, and thus Node 1 selects Node 3. Node 1 sends a message to Node 3 informing that it is selected. Node 3 then finds the next sensor, which is Node 5. After being informed by Node 3, Node 5 selects 6. Similarly, Node 6 selects Sensor 9. When Node 9 knows that it is selected and realizes that it has a neighbor (Node 1) already selected, it stops the search. Finally, a minimum cover $\{1, 3, 5, 6, 9\}$ is identified. Although no sensor knows the whole minimum cover, each of them knows whether it is in the cover or not and can send its image accordingly.

It is also worth noting that the mechanism works if there are more than one default members. All of them will start searching for the next sensor. Once a selected node detects that it has a neighbor already selected, it stops the search. It is possible that there is no default member. If this is the case, those sensors with the smallest starting angle can try to invoke the process, after they do not hear anything after some time. To reduce overhead, we can restrict only one of them start the searching. Nevertheless, the cover found may not be minimum.

When the sensors cannot cover $360°$, the sensors at the boundary can identify that. For example, refer to Figure 7, both Node 6 and Node 7 know that there is a gap of angle coverage since they both do not have a neighbor in the anticlockwise direction. In this case, the node which has a larger ending angle should start the search. If there is no default member, the search will end at a sensor that is at the other end (Sensor 4 or 5 in Figure 7). If there is a default member (Node 1 in Figure 7), the default member can pick up the search when it receives the message if it has not started its own search process. In our example, minimum set $\{1, 3, 5, 7\}$ is identified.

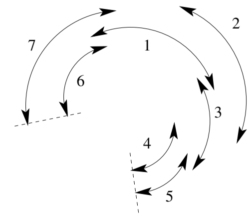

Fig. 7. Example of minimum set

It is not difficult to see that the message overhead is very small. After knowing neighborhood information, the only messages passing around would be for selected sensors to be included in the minimum cover. On the other hand, the proof of correctness still applies in the distributed scenario. That is, our distributed algorithm is able to find the minimum cover if one exists.

V. Simulation

In this section, we present the simulation results of our protocol. We would like to study the relationship between the size of minimum cover, image resolution and FOV. The simulation results are generated using MATLAB. The whole network area is divided into 20×20 grids, and the width of each grid is representing one unit distance. There is at most one camera node in each grid and the probability that a grid has a sensor depends on the density and is generated randomly. In our simulation, the probability is set to be 0.8. The orientations of all camera nodes are assigned randomly and the size of the raw images captured by each camera is 512×512. We assume that the object of interest is in cylindrical shape with radius $R_o = 5$ units and it is located at $(10, 10)$.

Any nodes with node distance less than or equal to 8 units are the sensors that will be able to capture images of the object of interest. 30 topologies are generated and each topology is simulated with 7 different requested image resolutions and 3 different FOVs. Each point in the following simulation results is the average value of 30 topologies. In our simulations, the image resolution κ is defined as the number of pixels required to represent one unit distance in the image.

Figure 8 shows the relationship between the requested image resolution and the number of candidates involved (number of sensors remaining after the first phase of our protocol in Section IV-A). It can be observed that the number of candidates declines as the resolution requirement increases. Additionally, when the FOV is larger, there would be fewer candidates. It is because when the FOV is larger, sensor has to be closer to the object in order to obtain a high resolution image. In figure 9, it can be observed that the angle coverage is

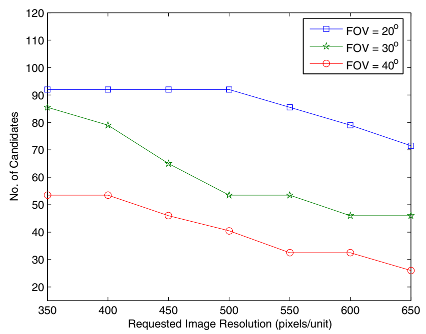

Fig. 8. R_o = 5 units, max. node distance = 8 units

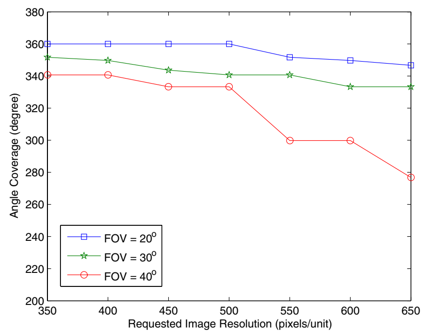

Fig. 9. R_o = 5 units, max. node distance = 8 units

getting poorer when the FOV is larger or the image resolution requirement is higher. Under a particular resolution level, larger FOV would have fewer candidates. Similarly, if we require a higher resolution, there will be fewer candidates and thus more difficult to cover $360°$.

Figure 10 shows the number of selected images under different κ and different FOV. Since the candidate set of a higher κ is a subset of the candidate set of a lower κ, it is expected that the size of minimum cover should increase with κ. We observe this trend when FOV $= 20°$. The reason why this is not the case when FOV $= 30°$ and FOV $= 40°$ is because a cover cannot be formed when κ is relatively large. It can be observed that there is a decreasing trend when the angle coverage is less than $340°$. As the images are covering a narrower view, fewer images will be needed. The drop can be regarded as an indication of a threshold value of the requested image resolution with satisfactory angle coverage.

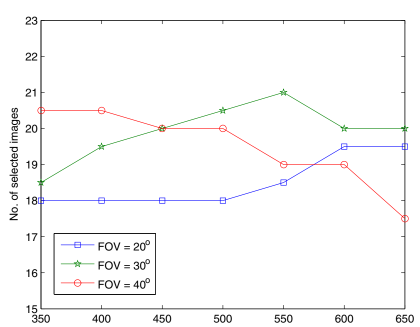

Fig. 10. R_o = 5 units, max. node distance = 8 units

VI. Conclusion

In this paper, the problem of angle coverage in visual sensor network is studied. We aim at maximizing angle coverage of the object of interest with minimum number of images to be sent. We develop a distributed algorithm with small overhead to obtain the minimum cover while fulfilling the image resolution requirement. The algorithm was analyzed through extensive simulations, and it can find the minimum cover set effectively. Depending on different settings, we may find the image resolution threshold value. The user should request for images with resolution lower than the threshold value in order to obtain a satisfactory angle coverage.

References

[1] G. J. Pottie and W. J. Kaiser, "Wireless integrated network sensors," in *Communications of the ACM*, vol. 43, no. 5, 2000, pp. 51 – 58.

[2] D. Estrin, D. Culler, K. Pister, and G. Sukhatme, "Connecting the physical world with pervasive networks," in *IEEE Pervasive Computing*, 2002, pp. 59 – 69.

[3] A. Mainwaring, J. Polastre, R. Szewczyk, D. Culler, and J. Anderson, "Wireless sensor networks for habitat monitoring," in *ACM International Workshop on Wireless Sensor Networks and Applications*, September 2002.

[4] C.-F. Huang and Y.-C. Tseng, "The coverage problem in a wireless sensor network," in *ACM International Conference Wireless Sensor Networks and Applications (WSNA)*, 2003, pp. 115 – 121.

[5] B. Cărbunar, A. Grama, J. Vitek, and O. Cărbunar, "Coverage preserving redundancy elimination in sensor networks," in *IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, 2004.

[6] X. Wang, G. Xing, Y. Zhang, C. Lu, R. Pless, and C. Gill, "Integrated coverage and connectivity configuration in wireless sensor networks," in *the First International Conference on Embedded networked sensor systems*, 2003.

[7] H. Gupta, Z. Zhou, S. R. Das, and Q. Gu, "Connected sensor cover: Self-organization of sensor networks for efficient query execution," in *IEEE/ACM Transactions on Networking*, vol. 14, no. 1, February 2006, pp. 55 – 67.

[8] S. Soro and W. B. Heinzelman, "On the coverage problem in video-based wireless sensor network," in *Workshop on Broadband Advanced Sensor Networks (BaseNets '05)*, Oct 2005.

[9] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Achieving 360° angle coverage with minimum transmission cost in visual sensor networks," in *IEEE Wireless Communications & Networking Conference*, March 2007.

[10] B. W. Parkinson, J. J. S. Jr., P. Axelrad, and P. Enge, "Global positioning system: Theory and application, Volume I," in *Progress in Astronautics and Aeronautics Series, Volume 163*, 1996.

[11] D. Niculescu and B. Nath, "Ad hoc positioning system (aps)," in *IEEE Globecom*, 2001, pp. 2926 – 2931.

[12] H. Lee and H. Aghajan, "Vision-enabled node localization in wireless sensor networks," in *COGnitive systems with Interactive Sensors (COGIS)*, March 2006.

[13] C. McCormick, P.-Y. Laligand, H. Lee, and H. Aghajan, "Distributed agent control with self-localizing wireless image sensor networks," in *COGnitive systems with Interactive Sensors (COGIS)*, March 2006.

[14] X. S. Zhou, I. Cohen, Q. Tian, and T. S. Huang, "Feature extraction and selection for image retrieval," in *ACM Multimedia 2000*, 2000.

[15] B. Bhanu and J. Peng, "Adaptive integrated image segmentation and object recognition," in *IEEE Transactions on Systems, Man and Cybernetics*, vol. 30, 2000, pp. 427 – 441.

[16] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," in *ACM Mobicom*, 2000, pp. 243 – 254.