This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

Achieving 360° Angle Coverage with Minimum Transmission Cost 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 network. We consider a tracking system where an object of interest moves around the network, and the sensors surrounding it are responsible for capturing the images of it. We aim at finding the *minimum cost cover* which preserves all the angles of view with minimum transmission cost. We proved formally that the minimum cost cover problem can be transformed into the shortest path problem. Due to the acyclic nature of graphs generated in the transformation, we can develop a distributed algorithm to solve the problem, which is a lot more efficient than the distance vector protocol. Our simulation results show that our algorithm can successfully save a lot of energy.**

I. Introduction

Wireless sensor networks have attracted tremendous research interests due to its vast potential applications such as surveillance, habitat monitoring, target tracking, disaster management, etc [1], [2]. Each sensor is powered by battery which is unlikely to be rechargeble, this energy limitation puts extra constraint in the operation of a sensor.

Different from conventional sensor networks, all the nodes in visual sensor networks are equipped with cameras and the data to be collected and transmitted are visual data. The size of visual data is far larger than the size of common measurement such as temperature, humidity, light intensity, etc. As a result, the transmission load of visual sensor networks should be minimized in order to prolong system lifetime. Additionally, conventional sensor networks often assume the omnidirectional sensing model [3]. However, this is not common in visual sensor networks. Every camera node has a particular capturing direction. Many approaches in omnidirectional sensor networks are not applicable to visual sensor networks.

Coverage problem is a fundamental issue in wireless sensor networks [4],[5]. It determines how well a target or area is tracked or monitored by sensors. In this paper, we study the *angle coverage* problem instead of the traditional area coverage problem.

In a tracking system, there is an object of interest moving around the network and the nodes around the object are responsible to capture the images of it. Due to the randomness of camera nodes distribution and oritentations, the images captured cover different sides of the object. A trivial way to preserve all the angles of view is to collect all the images, but this is not energy efficient. Since the nodes are close to each other, the visual data are highly correlated. An image is said to be redundant if the removal of it does not affect the angle coverage. In order to save energy, only selected images will be sent to the sink. Our goal is to find a set of sensors which requires least amount of transmission energy while preserving all the angles of view. We formally prove that this problem can be transformed into a well-known problem, *the shortest path problem* [6], which can be solved by the Dijkstra's algorithm. Due to some specific nature of the problem, an efficient distributed algorithm can be developed. Our distributed algorithm is far more efficient than the traditional distance vector protocol.

The rest of this paper is organized as follows: Section II presents the related work. Section III presents the problem statement and the simulation results are shown in Section IV. We finally conclude our paper in Section V.

II. Related Work

Coverage problem is an important issue in sensor networks. There are many research activities studying this problem [5], [7], [8]. Ref [9] also studies the coverage problem but specifically in video-based sensor networks. 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 cameras capture data, the traditional algorithm does not give expected results in terms of coverage preservation.

In a visual sensor network, it is not common that all the cameras monitor the area in one plane as the case in [9]. Usually, the sensors are distributed randomly in the network with arbitrary orientations. In our previous work [10], we developed an algorithm to find a **minimum cover** which preserves all the angles of view with minimum number of images. However, minimum number of images is different from minimum energy needed since different images may have to travel different distances to the sink. In this paper, we aim at finding a *minimum cost cover*, a set of images that achieves 360° coverage and requires the least transmission energy.

III. Network Model

We consider a sensor network where sensors are randomly distributed in the network. The main task of the sensors is to take images of a certain targeted object and transmit them to the sink upon request. All the sensors are equipped with

1525-3511/07/$25.00 ©2007 IEEE

cameras and identical in terms of intrinsic parameters such as field-of-view (FOV), focal length, etc. Due to the limited size of sensor, it is not capable of pan, tilt or zoom. Each camera node knows its physical location and orientation by means of image-based localization algorithms such as [11], [12]. The sensors that lie within the transmission range of sensor i would be the neighbors of i, they can communicate with each other.

A. Capture Range

For simplicity, we approximate the object of interest to be in cylindrical shape with radius R_o. As illustrated in Figure 1, ϕ is the angle of view covered by the sensor node and θ is the FOV of the node. Since every sensor node knows it physical location and orientation, R_1 is known. Given R_1, R_o and θ, ϕ can be found by trigonometry.

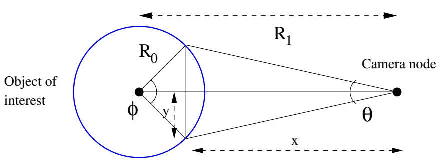

Fig. 1. Relationship between object of interest and sensor

Suppose the raw image size of the camera node is 512×512. Referring to the example in Figure 1, the full view of the captured image is occupied by the object of interest, $2y$ metres long visual data are projected onto 512 pixels horizontally. We may say that the image resolution is $\frac{512}{2y}$. As the camera node is farther away from the target, y increases, the camera can capture the object to a greater extent and thus 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 there is a tradeoff between image resolution and node distance.

B. Transmission Cost

Given a set of sensors S, let the angle of view covered by sensor node $i \in S$ be $[s_i, t_i]$. A set $V \subseteq S$ is a *cover* if for each angle $\beta \in [0°, 360°)$, there exists a sensor i in V such that $\beta \in [s_i, t_i]$. A cover C is a minimum cover if it is smallest in size among all covers. Let $|I_i|$ be the image size of node i, $hc(i)$ be the hop count of node i and E_t be the energy needed in transmitting one byte. The energy consumption in transmission of the each selected node will be

$$T(i) = |I_i| \times hc(i) \times E_t \qquad (1)$$

Intuitively, sending the minimum number of images seems to be the most energy efficient way. However, this is not true. For simplicity, we may assume that all the image sizes are about the same, $|I|$. As illustrated in Figure 2, each arrow is representing the capture range of one sensor and the number in circle is representing the hop count of each node. {1, 3, 5, 6, 9} is a minimum cover and its transmission cost is:

$$T_{total} = (3+3+4+4+4) \times |I| \times E_t \qquad (2)$$

$\{1, 2, 3, 4, 7, 8\}$ is another cover and its tranmission cost is:

$$T_{total} = (3+2+3+3+3+3) \times |I| \times E_t \qquad (3)$$

It can be observed that $\{1, 2, 3, 4, 7, 8\}$ consumes less energy than $\{1, 3, 5, 6, 9\}$. It shows that selecting the minimum cover is not necessarily be the most energy efficient way to preserve the angle coverage of the object of interest.

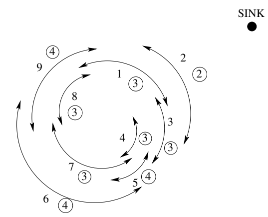

Fig. 2. Illustration of Distributed Version

IV. Problem Statement

We formally define the **minimum cost cover** problem as follows: Let C be the set of sensors that fulfill the image resolution requirement and possess images of the object of interest. For each $i \in C$, the angle of view covered by i, $V(i)$, is $[s_i, t_i]$ and the transmission load is $hc(i)$. We want to find a subset $M \subseteq C$ such that the total transmission cost is the minimum and M preserves 360^o view of the object of interest.

$$\text{minimize} \sum_{i \in M} hc(i) \text{ s.t. } \bigcup_{i \in M} V(i) = [0, 360°] \qquad (4)$$

A. Centralized Algorithm

We show that the problem can be transformed to the minimum cost path problem, which can be solved by the Dijkstra's algorithm. Given C, we construct a directed graph $G_C = (V, E)$ such that $V = C \cup \{S, T\}$. S and T are the source and destination of our minimum cost path problem, respectively. There are three types of edges:

1) Edges starting from S:
   $(S, i) \in E$ if $i \in C$ and $s_i = 0$
2) Edges going to T:
   $(i, T) \in E$ if $i \in C$ and $t_i = 360$
3) Edges linking nodes in C:
   $(i, j) \in E$ if $s_i < s_j \leq t_i$ and $t_i < t_j$

By defining E this way, a path from S to T first traverses a node i with start view angle 0, then goes to another node with overlapping angle of view with i, and keep on going until the path ends at a node with ending view angle as 360. In other words, the set of the nodes in the path is a cover.

Property 1: If $S \rightarrow n_1 \rightarrow n_2 \rightarrow \cdots \rightarrow n_k \rightarrow T$ is a path in G_C, $\{n_1, n_2, ..., n_k\}$ is a cover in C.

To find a minimum cost cover, we need to assign appropriate weights to the edges. The weights are:

1) $w(S, i) = hc(i)$ if $(S, i) \in E$
2) $w(i, T) = 0$ if $(i, T) \in E$
3) $w(i, j) = hc(j)$ if $(i, j) \in E$ and $j \neq T$

The weight of an edge is the transmission cost of the ending node of the edge. The cost of a path would be the sum of the transmission costs of the nodes on that path.

Property 2: The cost of path $S \rightarrow n_1 \rightarrow n_2 \rightarrow \cdots \rightarrow n_k \rightarrow T$ in G_C is the cost of cover $\{n_1, n_2, ..., n_k\}$ in C.

An example of the transformation is shown in Figures 3 and 4. The cost of the path $S \rightarrow 1 \rightarrow 2 \rightarrow 4 \rightarrow T$ is $hc(1) + hc(2) + hc(4)$.

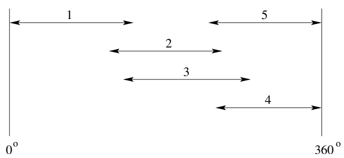

Fig. 3. Cover Example

Property 3: If M is a cover without redundant node (cover that all nodes are necessary to maintain the coverage requirement), then the nodes in M, in ascending order of starting angle, form a path from S to T.

Proof:

Suppose M is a cover without redundant node, $M = \{m_1, m_2, ..., m_k\}$ and the nodes are in ascending order of starting angle. Since M is a cover, it covers from 0° to 360° and thus $s_1 = 0$. There exists an edge between S and m_1. Obviously, $s_i < t_i$ and $s_i < s_{i+1}$. If $s_{i+1} > t_i$, there will be a gap between t_i and s_{i+1} and thus M will not be a cover, which leads to contradiction. Therefore, $s_i < s_{i+1} \leq t_i$. On the other hand, if $t_{i+1} \leq t_i$, node $(i+1)$ would be redundant as the view covered by it is completely overlapped by node i. Consequently, $s_i < s_{i+1} \leq t_i < t_{i+1}$. There exists an edge between m_i and m_{i+1}. We have a path $(S \rightarrow m_1 \rightarrow m_2 \rightarrow m_3 \rightarrow ... \rightarrow m_{k-1})$. As M is a cover, the ending angle of m_k is 360° and thus there is an edge between m_k and T. So, M forms a path from S to T. □

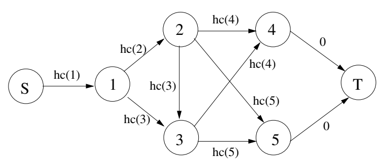

Fig. 4. Dijkstra's Algorithm Example

We now prove that finding a minimum cost cover in C is equivalent to finding a minimum cost path from S to T in G_C.

Lemma 1: If M is a minimum cost cover in C, then the nodes in M form a minimum cost path from S to T in G_C.

Proof:

Suppose to the contrary that $M = \{m_1, m_2, ..., m_k\}$ is a minimum cost cover in C but the nodes in M does not form a minimum cost path from S to T in G_C. Let $S \rightarrow n_1 \rightarrow n_2 \rightarrow \cdots \rightarrow n_k \rightarrow T$ be a shortest path. By Property 1, $N = \{n_1, n_2, ..., n_k\}$ is a cover. By Property 3, M forms a path from S to T. By Property 2, the cost of N is the cost of the path, which is smaller than the cost of M, which leads to contradiction. □

Lemma 2: M is a minimum cost cover in C if the nodes in M form a minimum cost path from S to T in G_C.

Proof:

Suppose to the contrary that M is a minimum cost path from S to T in G_C but the nodes in M does not form a minimum cost cover in C. Let $N = n_1, n_2, ..., n_k$ be a minimum cost cover. By Property 3, N forms a path from S to T. By Property 2, the cost of N is the cost of path $S \rightarrow n_1 \rightarrow n_2 \rightarrow \cdots \rightarrow n_k \rightarrow T$ in G_C, which is smaller than the cost of M, which leads to contradiction. □

From the above two lemmas, we can conclude that M is a minimum cost cover in C if and only if the nodes in M form a minimum cost path from S to T in G_C. Therefore, if a node, say the sink, knows C, it can construct G_C and apply the Dijsktra's algorithm to find the minimum cost cover.

The centralized algorithm works when there is a node with $s_i = 0$. If there is an image spanning across 0^o, this setting may not be appropriate. To solve this problem, we first determine whether there is any angle of view that is covered by one sensor, say i, only. In this case, i must be in any cover and we refer i as a *default member*. We then need to find a minimum cost cover that spans from t_i to s_i. If there is no default member, we can first identify the set of nodes that cover 0°. We assume each of the nodes in the set as a default member and find the relative minimum cost cover that contains that member. The cover with minimum cost is selected.

B. Distributed Algorithm

The Dijkstra's Algorithm is a centralized algorithm and may not be suitable in some situations. It is necessary for us to develop a distributed solution. The shortest path problem can be solved by the distance vector (DV) protocol in a distributed manner. However, DV requires a large message overhead and is not suitable for sensor networks. Fortunately, G_C is directed and acyclic. In acyclic graphs, nodes can be topologically sorted and shortest paths can be identified easier in a distributed manner [6].

Each sensor needs to obtain the angles of view of their neighbors only. If the angle of view covered by sensor i is $[s_i, t_i]$, j is a neighbor of i if $s_i \leq s_j \leq t_i$ or $s_i \leq t_j \leq t_i$. If $s_i < s_j \leq t_i$ and $t_i < t_j$, j is a *forward neighbor*, else j is a *backward neighbor*. The following summarizes the pseudo code of the distributed algorithm:

Distributed Algorithm

Default Node i

```
cost(i) = hc(i)
previous(i) = ⊥
total_cost = ∞
send ⟨cost(i), i⟩ to all forward neighbors

Upon receiving ⟨cost(w), w⟩ from backward neighbor w
{
    if total_cost > cost(w)
    {
        total_cost = cost(w)
        previous(i) = w
    }
    mark w
```

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the WCNC 2007 proceedings.

14: if all backward neighbors are marked
15: { send $\langle SELECT \rangle$ to $previous(i)$ }
16: }
Non-default Node n
17: $cost(n) = \infty$
18: $previous(n) = \perp$
19: Upon receiving $\langle cost(w), w \rangle$ from backward neighbor w
20: {
21: if $cost(n) > cost(w) + hc(n)$
22: {
23: $cost(n) = cost(w) + hc(n)$
24: $previous(n) = w$
25: }
26: mark w
27: if all backward neighbors are marked
28: { send $\langle cost(n), n \rangle$ to all forward neighbors}
29: }
30: Upon receiving $\langle SELECT \rangle$
31: { send $\langle SELECT \rangle$ to $previous(n)$ }

A default node starts the search process. It sends its cost to all its neighbors. Whenever a non-default node receives a cost from its backward neighbor, a new path is identified and cost is updated accordingly. A non-default node knows all the paths have been identified if all backward neighbors have sent it their costs. Then, it can advertise its cost, which is the minimum cost from S, to its neighbors. Since the graph is acyclic and all nodes can be topologically sorted, nodes can get information from backward neighbors one by one and there is no deadlock. Moreover, each node sends out at most two messages and the message complexity is far smaller than the DV protocol.

Referring to the example in Figure 4, *1* is a default member, it invokes the process. After getting the angles of view from neighbors, *1* knows that *2* and *3* are its forward neighbors. *1* then broadcasts $cost(1)$ to *2* and *3*. Initially, $cost(2) = \infty$. If $cost(2) > cost(1) + hc(2)$, *2* then updates $cost(2)$ to $cost(1) + hc(2)$ and $previous(2)$ to 1. Similarly, *3* updates $cost(3)$ and $previous(3)$ accordingly. Since *2* has only one backward neighbor, the updated $cost(2)$ is the least and is broadcasted to the forward neighbors (*3*, *4* and *5*). *3* receives $cost(2)$ and updates $cost(3)$ and $previous(3)$ accordingly. After that, *3* has received cost information from all of its backward neighbors. The updated $cost(3)$ is the cheapest and is broadcasted to *4* and *5*. *4* and *5* then update their costs in the same fashion. Since *4* and *5* end at 360° $(= 0^\circ)$, *1* with starting angle at 0° is their forward neighbor. *1* receives the cost information from both nodes, updates $cost(1)$ and $previous(1)$. The searching process can stop since *1*, the default node, has identified the minimum cost cover. *1* then informs node previous(1) that it is on the minimum cost path using the $\langle SELECT \rangle$ message. Suppose previous(1) = 4, *4* knows that it is in the minimum cost cover after receiving the $\langle SELECT \rangle$ message. It then informs previous(4) by sending a $\langle SELECT \rangle$ message. Eventually, all the nodes in the minimum cost cover are aware of their status.

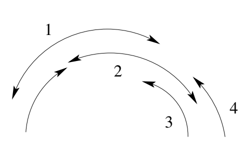

Fig. 5. Example without default member

It should be noted that the mechanism works fine when there are multiple default members. If there is no default member, the sensors that cover 0° may try to invoke the process, after they have not heard anything for some time. To reduce overhead, we can restrict only one of them starts the process. Nevertheless, the cover found may not be optimal. Besides, some timeout restrictions should be added to the algorithm when there is no default member. Referring to the example in Figure 5, suppose *2* is chosen to invoke the process. *2* then sends $cost(2)$ to *3* and *4*. Since *1* is also a backward neighbor of *3*, accordingly to the pseudo code, *3* would not send the updated cost until it has recevied cost information from all its backward neighbor (*1* and *2*). This would cause a deadlock to the whole process. To solve this problem, lines 14 and 27 should be modified as follows:

14 and 27: if (all backward neighbors are marked) or (timeout)

V. SIMULATION

The simulation results are generated using MATLAB. The whole network area is divided into 50×50 grids, and the width of each grid is representing one unit distance. Each grid contains at most one node with a randomly assigned orientation, and the probability that a grid has a sensor is 0.8. We assume that the object of interest is in cylindrical shape with radius $R_o = 4$ units. A node can capture images of a target if it is less than 8.5 units away from the target. Depending on applications, the users may request for images with different resolution. Only the sensors with images that fulfill the requested image resolution will be the candidates. Each point in the following simulation results is the average value of 20 topologies.

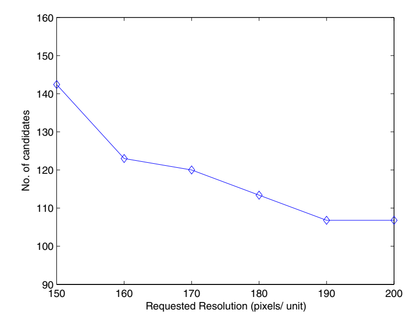

Fig. 6. Number of candidate, $FOV = 40^\circ$

Figure 6 shows the relationship between the requested image resolution and the number of sensors involved when $FOV = 40^\circ$. It can be observed that the number of candidates declines as the resolution requirement increases. Figures 7 and 8 show the comparison between Minimum Cover and Minimum Cost Cover. It can be observed that the size of Minimum Cover is always smaller than that of Minimum Cost Cover. However, there is a significant reduction in the sum of hop count when Minimum Cost Cover approach is applied. And the sum of hop count increases as the requested resolution increases. This is because the candidates of higher requested resolution is the

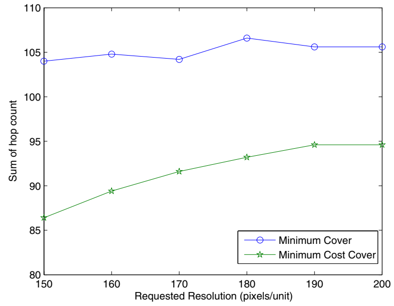

Fig. 7. Sum of hop count, $FOV = 40°$

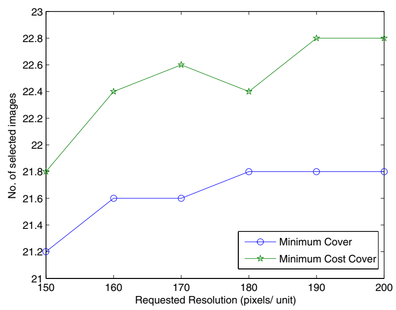

Fig. 8. Number of selected images, $FOV = 40°$

subset of the candidates of lower resolution. The Minimum Cost Cover approach can save about $10\% - 15\%$ transmission energy. Similar trends can be found when FOV varies, as shown in Figures 9 – 11.

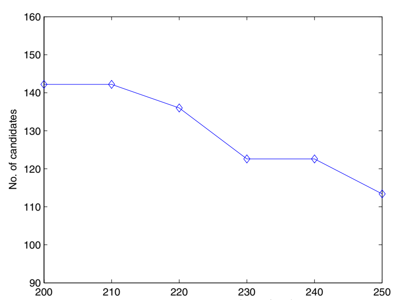

Fig. 9. Number of candidate, $FOV = 30°$

VI. Conclusion

In this paper, we study the angle coverage problem in visual sensor network. We formally proved that the minimum cost cover problem can be transformed into the shortest path problem and solved by Dijkstra's Algorithm. We develop a distributed algortihm to solve the problem which is more efficient than the distance vector protocol approach. The simulation results show that our algorithm can achieve a significant reduction in transmission cost.

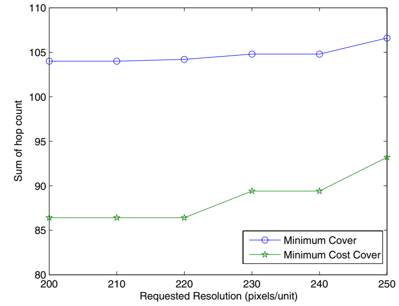

Fig. 10. Sum of hop count, $FOV = 30°$

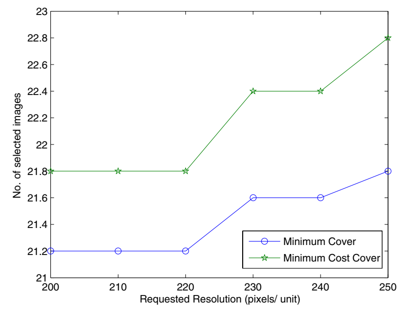

Fig. 11. Number of selected images, $FOV = 30°$

References

[1] F. Zhao and L. Guibas, *Wireless Sensor Networks : An Information Processing Approach*. Elsevier-Morgan kaufmann, 2004.

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

[3] H. Ma and Y. Liu, "On coverage problems of directional sensor networks," in *International Conference on Mobile Ad-hoc and Sensor Networks MSN*, December 2005.

[4] C.-F. Huang, Y.-C. Tseng, and L.-C. Lo, "The coverage problem in three-dimensional wireless sensor networks," in *IEEE Global Telecommunications Conference GLOBECOM*, vol. 5, 2004, pp. 3182 – 3186.

[5] 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.

[6] T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, *Introduction to Algorithms*. The MIT Press, 2001.

[7] Z. Zhou, S. Das, and H. Gupta, "Connected k-coverage problem in sensor networks," in *International Conference on Computer Communications and Networks, ICCCN*, 2004, pp. 373 – 378.

[8] 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.

[9] 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.

[10] K.-Y. Chow, K.-S. Lui, and E. Y. Lam, "Angle coverage in visual sensor networks," submitted to *ACM Transactions on Sensor Networks*, 2006.

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

[12] 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.