LISTEN: Non-Interactive Localization in Wireless Camera Sensor Networks

[1, 2]Yuan He, [3]Xingfa Shen, [1,2]Yunhao Liu, [4]Lufeng Mo, [3]Guojun Dai

[1]TNLIST, School of Software, Tsinghua University, China

[2]Dept. of Computer Sci. & Eng., Hong Kong Uni. of Sci. & Tech, Hong Kong

[3]Inst. of Comp. App. Tech. Hangzhou Dianzi Uni., China

[4]Dept. of Comp. Sci. & Tech. Xi'an Jiaotong Uni., China

Abstract

Recent advances in the application field increasingly demand the use of wireless camera sensor networks (WCSNs), for which localization is a crucial task to enable various location-based services. Most of the existing localization approaches for WCSNs are essentially interactive, i.e. require the interaction among the nodes throughout the localization process. As a result, they are costly to realize in practice, vulnerable to sniffer attacks, inefficient in energy consumption and computation. In this paper we propose LISTEN, a non-interactive localization approach. Using LISTEN, every camera sensor node only needs to silently listen to the beacon signals from a mobile beacon node and capture a few images until determining its own location. We design the movement trajectory of the mobile beacon node, which guarantees to locate all the nodes successfully. We have implemented LISTEN and evaluated it through extensive experiments. The experimental results demonstrate that it is accurate, efficient, and suitable for WCSNs that consist of low-end camera sensors.

1. Introduction

Wireless sensor networks (WSNs) have spread their uses across many different application fields in the past few years. More recently, the availability of low-cost hardware such as CMOS cameras and flash memory has fostered the development of wireless camera sensor networks (WCSNs) [1], which provide unprecedented advantages in a wide variety of applications [2, 3], such as environmental surveillance, safety guard, traffic management, battlefield, and person locator services.

Localization is a crucial issue in WCSNs, which involves determining the *locations* and *orientations* of camera sensor nodes. For many location-based services in WCSNs, their design correctness and effectivity are highly sensitive to the location accuracy of sensor nodes. Thus localization in WCSNs demands very high accuracy. Previous localization approaches for conventional WSNs are not applicable for WCSNs.

Many WCSN localization approaches have been proposed. We find they are all essentially *interactive*. In other words, the localization process largely relies on interactions among nodes to be located. Those approaches have the following drawbacks.

First, most existing approaches are hard to realize in practice. Some of them require expensive hardware, e.g. a robot or mobile objects with featured appearances [4, 5]. Some of them have ideal assumptions. The work in [6, 7] assume neighboring nodes always have overlapping FOVs. But practical deployments might be sparse. Nodes' FOVs do not necessarily overlap. Other approaches demand complex image processing [8], which are too resource-consuming to be applicable on low-end sensor nodes.

Second, interactive localization is vulnerable to malicious behavior such as sniffer attack [9]. In the mobile-assisted localization scenarios, the information of locations where the mobile object receives controlling signals from the camera sensors actually implies the FOVs (Field Of View) of the cameras. When a malicious sniffer masquerades as a normal mobile object and sniffs in the network, it is able to capture the information of the monitored area of a whole WCSN. Note that in many security and military applications, the locations of camera sensor nodes and their monitored FOVs are highly confidential information. Leakage of such information could lead to loss of property and even threats to human safety.

Last but not least, most interactive approaches require every node to capture many images in order to locate itself. Note that the power consumption of image sensing is much higher than that of scalar sensing (e.g. to sense temperature or humidity) [10]. Interactive localization thus inevitably consumes excessive energy, reducing the lifespan of a WCSN.

1052-8725/10 $26.00 © 2010 IEEE
DOI 10.1109/RTSS.2010.15

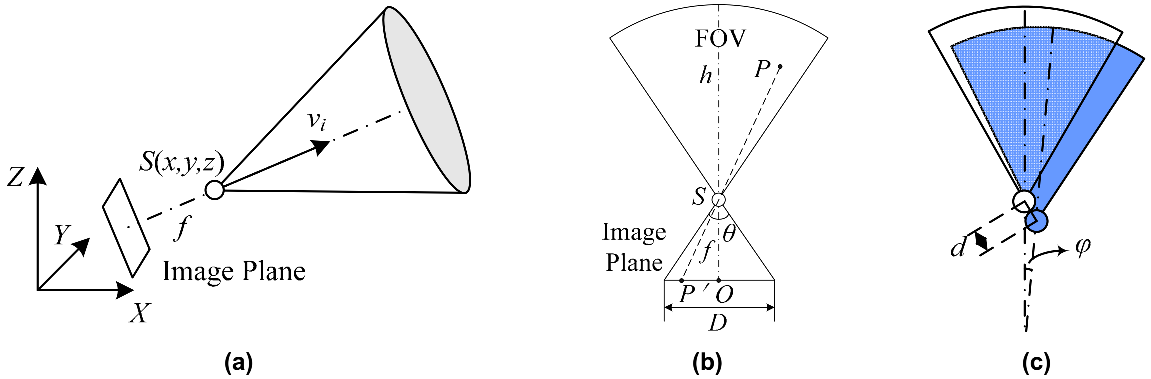


Figure 1. (a) FOV of *S*(*x*, *y*, *z*) based on the pinhole model. v_i is the orientation vector of *S*; *f* is the focal length; *θ* is the angle of view. (b) The 2D pinhole model. *h* is the depth of view; *D* is the image plane. (c) Comparison betewen the real (transparent) and estimated (blue) FOVs.

Our work is motivated by the need of location information of camera sensors in GreenOrbs [11]. GreenOrbs is a large-scale sensor network system that supports a wide variety of applications [12]. Camera sensors are deployed in GreenOrbs to enable fire detection and rescue in the wild forest. It is a challenging task for GreenOrbs to accurately locate the camera sensors. GPS devices fail to work under the dense tree canopy. It is also too expensive to equip GPS with every sensor node. Besides, conventional localization approaches for WSNs yield unsatisfactory results in the complex forestry environments, due to the signal irregularity and environmental dynamics.

To address the above issues, in this paper we propose LISTEN, a non-interactive localization approach for WCSNs. LISTEN employs a mobile beacon node with very simple appearance. The whole localization process does not require any interactions among the nodes. When the mobile beacon node traverses the deployment area, every node only needs to silently listen to the beacon signals and capture a few images until successfully locating itself. Our main contributions are summarized as follows.

First, we propose the non-interactive light-weight localization approach, LISTEN. To locate itself, every node needs only a few times of image sensing and light-weight image processing. LISTEN has no complicated requirements on hardware or specific assumptions on the network deployments.

Second, we design the trajectory of the mobile beacon node, which guarantees successful localization of all the nodes. A node to be located only needs to receive beacon messages but sends nothing, thus avoiding leakage of its location information.

Third, we have implemented LISTEN on the camera sensor nodes produced by ourselves. The experimental results demonstrate that LISTEN outperforms other approaches with high accuracy and consistent performance, using only commercial off-the-shelf devices.

The remainder of this paper is organized as follows. Section 2 briefly introduces the background of localization in WCSNs and discusses the related work. Section 3 elaborates on the design of LISTEN, followed by the proofs of correctness in Section 4. Section 5 presents the implementation and the experimental results. We conclude in Section 6.

2. Background and Related Work

2.1 Localization in WCSNs

WCSNs differ from conventional WSNs with some distinct characteristics. Generally, multimedia data occupy much more memory storage on the sensor nodes. The available network bandwidth in a WCSN, however, is rather limited. Real-time data collection is already a challenging issue in conventional WSNs [13, 14], not to mention the data collection in WCSNs. Besides, the power consumption of sensing once on a camera sensor (i.e. capture an image) is much higher than that on a scalar sensor, such as thermometer and manometer. Such facts necessitate innovative designs of localization, sensing control and coordination, data collection, routing, and query processing techniques [2, 3, 15-18].

Sensing model. Many existing works in conventional WSNs assume disk-based sensing [19], while WCSNs employ the directional sensing model. The field of view (FOV) of a camera sensor is usually based on the pinhole model and shaped as a cone in 3D space or a sector in 2D plane, as shown in Figures 1(a) and 1(b) respectively. The FOV of a camera sensor is determined by the camera's extrinsic parameters (namely location and orientation) and intrinsic parameters (including focal length, image format, and principal point, etc.). Generally the intrinsic parameters are fixed for the camera sensors.

Localization of a camera sensor refers to determining its location and orientation. In Figure 1(a), locating node *S* means determining its coordinates (*x, y*, *z*) and unit orientation vector v_i. To function effectively, camera sensors demand very accurate localization. Figure 1(c) shows an illustrative example in the 2D plane. The tiny estimation errors of the location (*d*) and the orientation (φ) can result in a significant estimation error of the FOV. Specifically in Figure 1(c), less than 80% of the real FOV is covered by the estimated one. The relative error is over 20%.

2.2. Related Work

As far as we know, conventional localization approaches in WSNs yield errors of one meter at least [20-22], which is unacceptable for localization in WCSNs. GPS-based solutions are too costly and mainly function in outdoor environments. Range-based approaches like TOA, TODA, and AOA provide better accuracy but require extra hardware support [23].

As we mentioned in Section 1, the existing approaches of localization in WCSNs are all interactive, which are classified into two main categories: collaboration-based and mobile-assisted. The collaboration-based approaches borrow the idea from the field of computer vision [8, 24]. Nodes locate themselves by collaboratively interpreting the common visual information in their overlapping FOVs. Mobile-assisted approaches employ mobile objects (robots or beacon nodes) to assist the localization process. It is assumed the mobile objects always know their own coordinates, have distinctive appearance, and can be controlled by other nodes via wireless controlling signals.

D. Devarajan *et al*. [8] address the issue of calibrating distributed cameras. Each node independently calibrates itself based on information shared by the nodes adjacent to it in the vision graph. Nodes with overlapping FOVs collaboratively interpret the common view among themselves. It does not assume any single node knows the global configuration of the entire network. As a result of calibration, each node has an estimate of its own location and orientation.

Similar with the above scheme, A. B.-Sweeney *et al.* [6] propose another collaboration-based approach. They assume a WSN includes camera sensors and a number of normal sensor nodes. The normal sensor nodes identify themselves to camera sensors using modulated LED emissions. Pairs of camera sensors exchange information about the normal nodes observed in their FOVs to compute their relative rotation and translation matrices. H. Lee *et al.* [7] propose to trigger simultaneous image sensing by different camera sensors to capture a passing target. A small number of such joint observations may help to construct a relative coordinate system among those camera sensors.

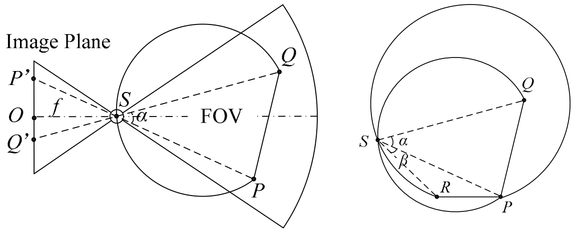
The above schemes have apparent limitations in the context of WCSNs. For example the scheme in [8] requires the nodes to form local calibration clusters, each of which has a minimum of three nodes with eight common scene points. Schemes in [6] and [7] require camera sensors to identify common objects in their FOVs. The practical WCSNs, however, are very unlikely to satisfy such requirements. Camera sensors are often ad-hoc and sparsely deployed, without dense clusters or overlapping FOVs among them. As a result, many nodes still cannot locate themselves using these schemes. The process to extract featured points or common objects from the images is also too costly for low-end camera sensors motes.

X. Liu *et al*. [25] propose a self-calibration protocol that does not guarantee successful localization for all the nodes in a WCSN. In [4] the authors propose a scheme of robot-assisted localization. In their targeted scenarios, camera sensors are deployed on the same plane (e.g. the ceiling of a room), which is parallel to the robot's motion plane (e.g. the floor). Their solution only addresses localization in the 2D plane. The robot knows its own coordinates and acts as a reference object in the captured images. The sensor network topology is modeled as a forest. Camera sensors in the same tree collaborate with each other to control the patrolling routes of the robots. In this way, every node may obtain sufficient observations to localize itself. When a tree of sensors is localized, the root node of that tree initiates a complicated process to instruct the robot to discover other adjacent trees. Both the localization and discovery processes incur large amount of communication cost on the sensors and the robot. The total energy consumption is easily affected by the node densities and network topologies. When the camera sensors are sparsely deployed and the entire network consists of many weakly-connected components, energy drains even faster because the discovery processes are triggered frequently.

3. Design of LISTEN

3.1 Assumptions

First we assume the intrinsic parameters of all the camera sensors are identical, fixed, and known before-



(a) (b)

Figure 2. Localization of node *S* in the 2D plane. (a) The beacon node and its images on the image plane. (b) Locate *S* as the intersecting point of two arcs

hand, because we generally use the same type of camera sensors in a WCSN.

Second, we assume the communication range of the sensor node is longer than the depth of the camera's FOV. This is usually true in practice. For example, the communication range of CC2420 radio using the maximum transmission power is 100 meters in an outdoor environment [10], while a distance of 100 meters is in general longer that what needs to be monitored by a single camera sensor. Therefore, a beacon node and a camera sensor can communicate with each other as long as the former is in the FOV of the latter.

As for the mobile beacon used in LISTEN, it is a common sensor mote enabled with mobility. It always knows its own coordinates, as many existing proposals assume [4, 20]. A featured tag (e.g. LED or a piece of colored paper) is attached to the beacon node, so that the beacon node is easily identified by camera sensors and differentiated from the surroundings.

3.2 2D Localization

A mobile beacon node is employed to assist localization, which traverses the deployment area of a WCSN, passing a set of *beacon positions*. At each beacon position, it broadcasts a beacon signal that includes its current coordinates. On receiving the signal, a camera sensor captures an image and then tries to extract the featured tag of the beacon node from the image. We name the image of the featured tag as *beacon image*. The coordinates of beacon images are then used to calculate the angular distances between pairs of beacon positions, e.g. $\angle PSQ$ in Figure 2(a). In the next subsections, we elaborate on how to utilize such angular information to locate the camera sensors.

3.2.1 The Localization Scheme

First we introduce the algorithm of LISTEN when all the nodes including the mobile beacon are in the same 2D plane. The orientation vectors of the cameras are in the plane too.

As shown in Figure 2, *S* is a camera sensor. Abusing notations, we use $S(x_0, y_0)$ to denote the camera pinhole. *O* is the projection of *S* on the image plane, i.e. the image center. Suppose points $P(x_1, y_1)$ and $Q(x_2, y_2)$ are two beacon positions where the beacon node broadcasts beacon signals. *P'* and *Q'* are the corresponding beacon images. $\angle PSQ$ is called the *angular distance* between beacon positions *P* and *Q*. We have

$$\begin{aligned}\angle\alpha &= \angle QSP = \angle Q'SP' = \angle Q'SO + \angle OSP' \\ &= \tan^{-1}\left(\frac{|OQ'|}{|OS|}\right) + \tan^{-1}\left(\frac{|OP'|}{|OS|}\right) \\ &= \tan^{-1}\left(\frac{|OQ'|}{f}\right) + \tan^{-1}\left(\frac{|OP'|}{f}\right) \qquad (1)\end{aligned}$$

Note that *f* is known according to the intrinsic parameters. |*OQ'*| and |*OP'*| can be measured from the captured image. $\angle\alpha$ can be determined. According to Theorem 1 (refer to Section 4), *S* lies on arc $\overset{\frown}{QP}$ whose circumferential angle is $\angle\alpha$. Thus we have

$$\frac{\overrightarrow{SP}}{|SP|}e^{i\alpha} = \frac{\overrightarrow{SQ}}{|SQ|} \quad (2)$$

Similarly, by introducing a new beacon position *R*, we draw another arc $\overset{\frown}{PR}$ whose circumferential angle is $\angle\beta$, as shown in Figure 2(b). Node *S* lies on $\overset{\frown}{PR}$.

$$\frac{\overrightarrow{SR}}{|SR|}e^{i\beta} = \frac{\overrightarrow{SP}}{|SP|} \quad (3)$$

Thus *S* is located as an intersecting point (the other is *P*) of arcs $\overset{\frown}{QP}$ and $\overset{\frown}{PR}$. Solving Formulas (2) and (3) generally yields the unique coordinates of *S*.

In two special cases, however, the coordinates of *S* cannot be uniquely determined: (1) *P, Q, R*, and *S* are collinear; (2) *P, Q, R*, and *S* are concyclic. Any point on their common line (or arc) is an eligible solution. In Section 3.2.2, we present the design of the mobile beacon trajectory to guarantee unique localization.

Now we continue to calculate the orientation of node *S*, which is denoted by the direction of vector $\overrightarrow{OS}$ in Figure 2(a). Recall that *f* and the coordinates of *S* and *P* are already known, while |*OP'*| can be measured from the captured image. Thus the unit orientation vector of *S* is calculated as follows.

$$\frac{\overrightarrow{OS}}{|OS|} = \frac{\overrightarrow{SP}}{|SP|}e^{i\sin\angle P'SO} \text{ where } \angle P'SO = \tan^{-1}\left(\frac{|OP'|}{f}\right) \qquad (4)$$

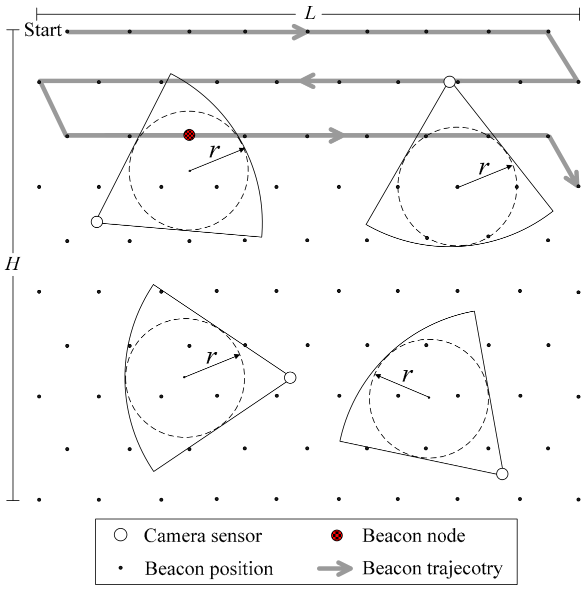


Figure 3. The mobile beacon trajectory

3.2.2 **The Mobile Beacon Trajectory**

According to the above scheme of LISTEN, we obtain the sufficient and necessary condition for a node to be uniquely located in the 2D plane: *The node has captured three beacon images and the three beacon positions are not collinear or concyclic with it.*

Recall that the localization process using LISTEN is completely non-interactive. The mobile beacon node doesn't have any prior or posterior knowledge of the other nodes' FOVs. When traversing the deployment area, the beacon node doesn't know whether a beacon position is captured by any other node. Hence, it is non-trivial to satisfy the above sufficient and necessary condition.

This subsection presents the design of the mobile beacon trajectory. As long as the mobile beacon broadcasts beacon signals at beacon positions along the designed trajectory, it is guaranteed that every node in the WCSN, no matter where it is, can be uniquely located.

We assume all the nodes as well as their FOVs are included in a rectangle area without any obstacle. Recall that all camera sensors are assumed to have identical intrinsic parameters. We use h and θ to denote the depth and opening angle of the FOV, respectively. Let r denote the radius of the inscribed circle of the cameras' FOV. We have

$$r = \frac{h}{(1 + 1/\sin\frac{\theta}{2})} \quad (5)$$

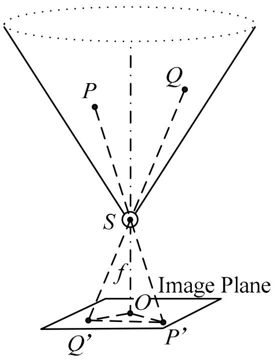


Figure 4. Calculation of angular distance between 3D beacon positions

Figure 3 shows the beacon trajectory in our design. Then the deployment area is partitioned into numerous equilateral triangles, whose side lengths are all equal to r. We select the vertices of the triangles as beacon positions. The beacon node starts from the upper left corner point, moves along the trajectory, and broadcasts its coordinates at those beacon positions.

Using such a trajectory, every node is able to capture at least three beacon images, which correspond to three beacon positions on the plane that are neither collinear nor concyclic with the node itself. Theorem 3 in Section 4 presents the proof.

3.3 Extension of LISTEN to 3D Localization

This subsection presents the design of LISTEN in 3D space. We first explain the calculation of angular distances between beacon positions, followed by the discussion on the uniqueness of localization. We then propose a selection strategy of beacon positions that guarantees to locate an entire network of camera sensors in 3D space.

3.3.1 Calculation of angular distance

The angular distances between beacon positions in 3D space can be calculated similarly as that in 2D plane. Figure 4 shows an example. P and Q are two beacon positions. P' and Q' are the corresponding beacon images on the image plane of camera sensor S. The angular distance between P and Q, i.e. $\angle PSQ = \angle P'SQ'$. We have

$$\begin{cases} |SQ'|^2 = |SO|^2 + |OQ'|^2 = f^2 + |OQ'|^2 \\ |SP'|^2 = |SO|^2 + |OP'|^2 = f^2 + |OP'|^2 \\ |P'Q'| = |SQ'|^2 + |SP'|^2 - 2|SQ'||SP'|\cos\angle P'SQ' \end{cases} \quad (6)$$

Since $|OQ'|$ and $|OP'|$ can be measured from the captured image, solving Formula (6) yields $\angle P'SQ'$.

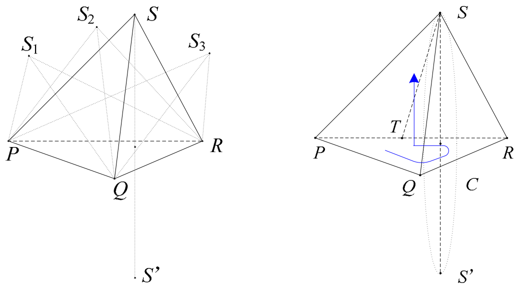


Figure 5. (a) Multiple solutions when using only three beacon positions; (b) The unique solution when using four beacon positions.

3.3.2 Uniqueness of 3D localization

Suppose $\angle PSQ = \angle \beta$, then S should be contained in a rotating surface in 3D space. This surface satisfies the following condition: Using any node K on this surface as reference, the angular distance between P and Q (namely $\angle PKQ$) is equal to $\angle \beta$.

Now we examine whether three beacon positions are sufficient to uniquely locate a node. Suppose P, Q, R are three different beacon positions. According to the result in the above paragraph, we have three different rotating surfaces. Any intersecting point of the three surfaces is a candidate position of sensor S.

As shown in Figure 5(a), using only three beacon positions generally yield more than one solution to the coordinates of S, namely S, S_1, S_2, S_3, and their counterpoints across the plane PQR (For clear display, we do not show all of them). For i=1, 2, 3, $\angle PSQ = \angle PS_iQ, \angle PSR = \angle PS_iR, \angle RSQ = \angle RS_iQ$. In other words, three beacon positions in 3D space are insufficient to uniquely locate a sensor node.

At least four beacon positions are required in 3D localization. For example in Figure 5(b), there is an additional beacon position T on line PR. Based on the angular distances among P, T, and R, one can locate node S on a circle (denoted by C). Line PR is perpendicular to the plane of C, and the center of C is on line PR.

On the other hand, S is located on another two rotating surfaces. One of them has PQ as its central axis. The other has QR as its central axis. Obviously, at least one of the two rotating surfaces does not contain circle C. There must be two intersecting points between C and the other two rotating surfaces. Let point S and its counterpoint S' denote the intersecting points. Now the number of candidate locations is reduced to two.

We further introduce a *right-hand rule* to filter out point S'. Specifically, the node decides a counter-clockwise sequence of the beacon positions P, T, and R, simply based on the node's view of the beacon images. In Figure 5(b), the sequence is $Q \rightarrow R \rightarrow P$. When fitting a right hand with the sequence, the pollex points to the actual position of the camera sensor.

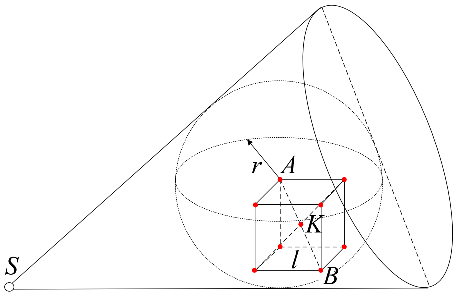


Figure 6. Beacon positions in 3D space

To sum up, in 3D space, generally a camera sensor S can be uniquely located by using four different beacon positions together with the right-hand rule. Meanwhile, the orientation of S can be calculated similarly as the case of 2D localization. We skip this part due to the limit of paper length.

3.3.3 Selection of beacon positions

According to the result in Section 3.3.2, we can use four beacon positions to uniquely locate a node as long as the four positions satisfy the following condition: *exactly three of them are collinear*. Such a condition serves as general guidance for selecting beacon positions in 3D localization.

Note that selection of beacon positions actually depends on the specific conditions of deployment area. Here we just propose an optional selection strategy.

We assume all the camera sensors and their FOVs are included in a cubic deployment area without any obstacle. Recall that the FOV of a camera sensor in the 3D space is a shaped as a cone. As shown in Figure 6, let r denote the radius of the inscribed sphere of node S's FOV. We use h and θ to denote the depth and opening angle of the FOV, respectively. In this case, Equation (5) still holds.

Further, the deployment area is partitioned with cubes whose side lengths are all equal to l, such that the diagonal length of a cube is r. We have

$$l = \frac{h}{\sqrt{3}(1 + 1/\sin\frac{\theta}{2})} \quad (7)$$

The vertices and the diagonal joins of the cubes are selected as beacon positions. It is easy to see that the FOV of an arbitrary camera sensor covers at least nine beacon positions, namely the eight vertices and the diagonal join of a cube. Because every pair of diagonal vertices (e.g. A and B in Figure 6) and the diagonal

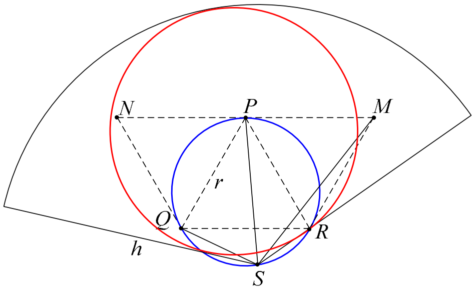


Figure 7. Uniqueness of localization

join (*K* in Figure 6) are collinear, there exist four covered beacon positions while exactly three of them are collinear. Using the beacon images corresponding to the four beacon positions, camera sensor *S* can uniquely locate itself.

4. Correctness

Theorem 1: (*The Theorem of Circumferential Angle*) *P and Q as two points on the plane and an angle* $\angle\alpha$ ($0 < \angle\alpha < \pi$). *Then the set of all the points S on the plane, which satisfies* $\angle PSQ = \angle\alpha$, *consists of two symmetric arcs where P and Q are the endpoints.*

Theorem 1 is a classical geometric theorem. We neglect the proof here. □

Theorem 2: *Let P, Q, and R be the three beacon positions covered by the FOV of a camera sensor S. When P, Q, R, and S are not collinear or concyclic, the coordinates of S can be uniquely determined by using LISTEN in the 2D plane.*

Proof: Suppose there is a point *S'* other than *S* which satisfies all the angle constraints. As the precondition of the proposition says, *P, Q, R,* and *S* are not collinear. Hence according to the scheme of LISTEN in 2D plane, *S'*, *S*, *P*, and *Q* are concyclic because $\angle PSQ = \angle PS'Q$. Likewise, *S'*, *S*, *P*, and *R* are concyclic. Note that two different circles have at most two intersecting points, so *S'*, *S*, *P*, *Q*, and *R* are concyclic. Obviously that is contradicted with the precondition "*P*, *Q*, *R*, and *S* are not concyclic."

Theorem 2 is proved. □

Theorem 3: *Using the beacon trajectory proposed in Section 3.2.2, every camera sensor in the 2D plane can be uniquely located.*

Proof: According to Theorem 2, proof of Theorem 3 is to prove a node's FOV covers three different beacon positions that are not collinear or concyclic with the node itself.

Figure 7 shows a camera sensor *S* and a part of the deployment area with selected beacon positions (i.e. *P, Q, R, M,* and *N*). Recall that the distance between adjacent beacon positions is *r*, which is equal to the radius of the inscribed circle of a FOV (red circle). Therefore, no matter where the center of the inscribed circle is, the circle covers an equilateral triangle whose vertices are three beacon positions. Without loss of generality, we suppose the covered triangle is ΔPQR.

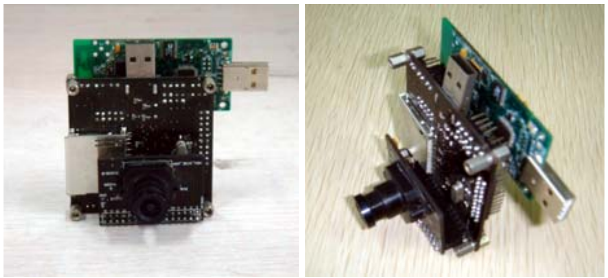


Figure 8. Camera sensor in implementation

First, *P*, *Q*, and *R* are obviously not collinear, so *P, Q, R,* and *S* are not collinear. Second, we draw the circumcircle of ΔPQR (blue circle). As long as *S* is not on the circumcircle, *P*, *Q*, *R*, and *S* are not concyclic. Hence *P*, *Q*, and *R* satisfy the constraints stated in Theorem 2 and *S* can be uniquely located.

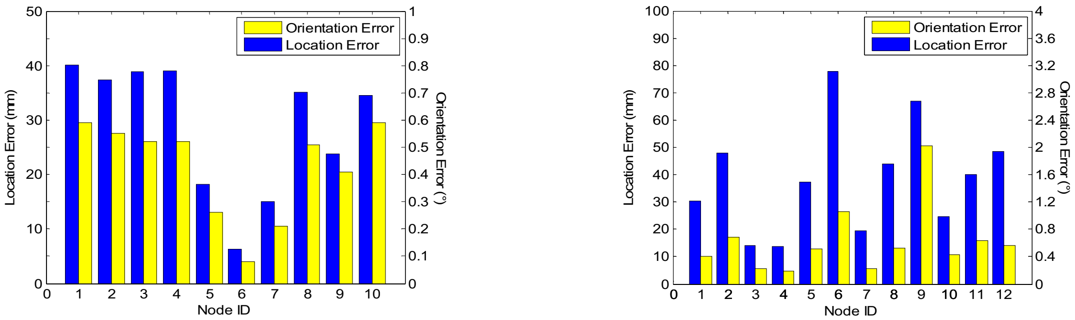
Otherwise, *S* is on the circumcircle of ΔPQR. Without loss of generality, we suppose *S* is at the position shown in Figure 7. $\angle RSQ = 120^\circ$. Clearly *MR* is tangent to the circumcircle of ΔPQR. Therefore, the FOV of *S* covers *M* if $|SM| \le h$. From Equation (5), we have $r<h/2$. According to the triangular inequality, $|SM|<|SR|+|MR|<|QR|+|MR|=2r<h$. Thus we prove *M* is covered by the FOV of *S*. $\angle RSP = 60^\circ$, so *P, M, R,* and *S* are not concyclic. Hence the three beacon positions *P*, *M*, and *R* satisfy the condition stated in Theorem 2 and *S* can be uniquely located.

To sum up, in all instances *S* can be uniquely located. Theorem 3 is proved. □

As for 3D localization, we have illuminated in Section 3.3.3 that a network of camera sensors can be uniquely located by using the selected beacon positions. We omit the detailed proof here.

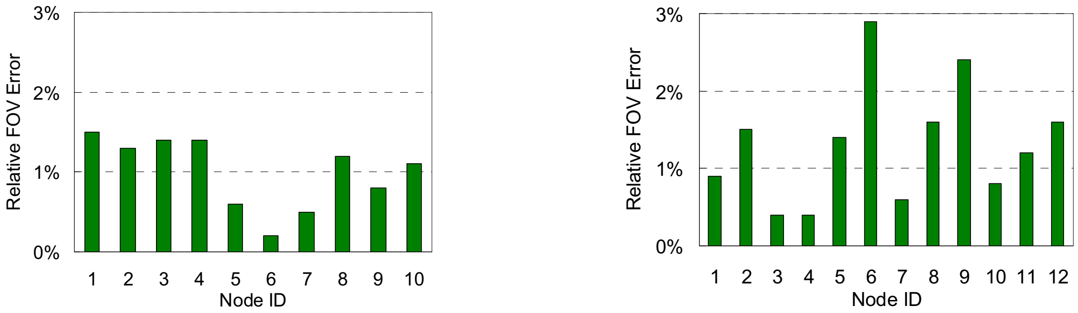
5. Implementation and Experiments

We have implemented LISTEN on our own-produced camera sensor motes. The hardware design conforms to the paradigm of CMUcam3 [3]. Every camera sensor is connected with a TelosB mote [10], as shown in Figure 8. For all the cameras, the depth and the opening angle of FOV are configured as 6 meters and 52º, respectively. The mobile beacon node (a TelosB mote) is mounted on top of a bracket. The red (or green) LED of the mote is kept on, which serves as the featured tag of the mobile beacon. We use three such beacon nodes simultaneously to speed up the localization process. As we observe, LISTEN



(a) Classroom experiment **(b) Corridor experiment**

Figure 9. Location and orientation errors in two experiments



(a) Classroom experiment **(b) Corridor experiment**

Figure 10. Relative FOV errors in two experiments

Table 1. Summary of the Experimental Results

	Location Error (mm)		Orientation Error (°)		Relative FOV Error (%)	
	Avg.	S.D.	Avg.	S.D.	Avg.	S.D.
PI	1,173	301.2	N/A	N/A	N/A	N/A
LISTEN (classroom)	28.82	12.08	0.42	0.18	1.00	0.45
LISTEN (corridor)	38.73	20.03	0.62	0.50	1.31	0.77
LISTEN (webcam)	23.31	7.57	0.45	0.13	N/A	N/A

has satisfactory performance in both 2D and 3D cases. In order to compare it with the conventional localization approaches that work in the 2D case, only results in the 2D case are shown here.

5.1 Comparisons

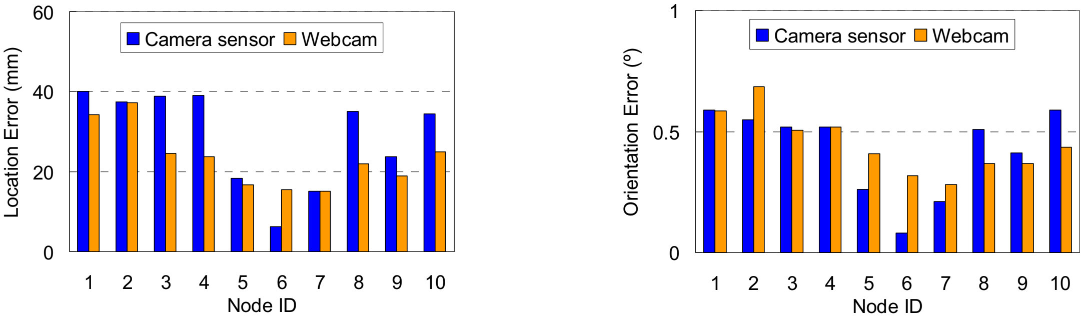
The first experiment compares LISTEN with PI [20], a state-of-arts mobile-assisted approach for localization in conventional WSNs. Instead of using the absolute values of RSSI, PI utilizes only the comparison relationship of the measured RSSI values between the mobile beacon and the other nodes to do localization. The resulting accuracy of PI is better than most existing localization approaches, as demonstrated in [20].

We conduct the experiments in a 12m×12m classroom. 10 sensor nodes are randomly deployed and located using LISTEN and PI, respectively. The results in Table 1 demonstrate that LISTEN apparently outperforms PI with much lower location error and more consistent accuracy. Moreover, the accuracy of RSSI-based localization is sensitive to various factors, such as signal fading, multipath, interference, and environmental dynamics. Image-based localization using LISTEN performs stably against such factors.

5.2 Evaluation in Different Environments

In this group of experiments, we evaluate LISTEN in two different environments. Other than the first deployment in the classroom, we further conduct another experiment with 12 nodes in a long corridor (6m×30m) in an office building. Due to the relatively long and narrow space in the corridor, the beacon positions used for locating a node are relatively far from the node. The resulting angular distances between beacon images are thus relatively small, which in turn results in larger relative errors.

We measure the location error, orientation error, and relative FOV error (the percentage of the real FOV that is missing in the estimated FOV) for all the nodes. Figures 9 and 10 plot the comparison results.



(a) Location errors in the classroom **(b) Orientation errors in the classroom**

Figure 12. Comparisons between the results on camera sensors and webcams

We can see that results in the classroom are more accurate and consistent than those in the corridor (please note the difference in Y-axis scales). This is mainly due to the difference in the angular distances between the beacon images, as we mentioned in the first paragraph of Section 5.2.

Moreover, the resulting averages of the relative FOV errors are only 1% and 1.31%, respectively. In other words, LISTEN is capable of supporting location-based services of WCSNs with very accurate localization.

It is also interesting to see that nodes 5, 6, 7 in the classroom and nodes 3, 4 in the corridor are more accurately located than the other nodes. We then go through and compare the images captured by all the nodes. The finding is that the beacon images captured by those five nodes are all near to the center of the image. Such beacon images correspond to lower distortions than the beacon images captured by the other nodes, hence resulting in smaller location errors.

The above finding indicates that using different subset of beacon images yields localization results with different accuracies. It may be feasible and beneficial to design a refining procedure with LISTEN, which intelligently selects the most appropriate subset of beacon images to achieve the best localization results. We will address this issue in our future work.

5.3 Impact of Image Quality

Figure 11 shows two pictures. The right one taken by a 300K-pixel webcam has obviously higher quality than the left one taken by our camera sensor. It could be a major concern that the low image quality of camera sensors might degrade the performance of localization using LISTEN. To further evaluate the impact of image quality on the localization accuracy of LISTEN, we have ported it to run over the webcam pictures.

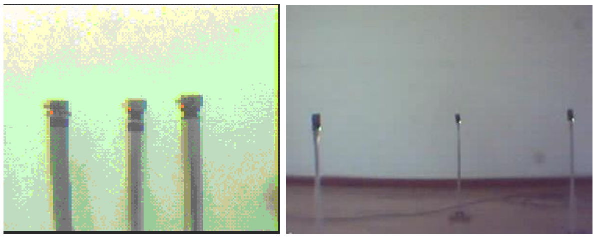


Figure 11. Images captured by camera sensor (left) and webcam (right)

We deploy a webcam at exactly identical locations and orientations as those camera sensors in the classroom and corridor experiment. Figure 12 compares the experimental results. Due to the page limit, we only present the detailed location and orientation errors in the classroom experiment, as shown in Figures 12(a) and 12(b). Interestingly, we find the accuracy of LISTEN is almost not affected by the image quality. The results on the webcam are only slightly better than those on the camera sensors.

Now we briefly summarize the experiments. LISTEN realizes very accurate localization in WCSNs. Compared to a conventional RSSI-based approach, LISTEN performs apparently better and more stably under various environmental settings. Moreover, the performance of LISTEN is robust against the impact of image quality. Thus it is suitable to localization of low-end camera sensors.

6. Conclusion

WCSNs present novel application fields of the WSN technology. Localization, although being well studied in the literature of WSNs, remains a challenging issue in WCSNs. Various approaches have been proposed but are all essentially *interactive*. Those approaches suffer vulnerability to malicious attacks, poor applicability, and excessive overhead.

This paper proposes LISTEN, non-interactive localization for WCSNs. LISTEN is energy-efficient and easy to implement in practice. By employing a mobile beacon with simple appearance to assist localization, every node to be located only needs to passively listen to the beacon signals and does not send any packet throughout the whole localization process. By calculating the angular distances between beacon positions, a node needs as few as three times of image sensing to locate itself.

In our future work, we will address the issue of location refining, as we mention in Section 5.2. We also plan to carry out large-scale implementation of LISTEN with our own-produced camera sensors in the GreenOrbs deployments.

Acknowledgment

This work is supported in part by NSFC/RGC Joint Research Scheme N_HKUST602/08, National Basic Research Program of China (973 Program) under Grants No. 2011CB302705, National Natural Science Foundation of China under Grant No. 60773042 and No. 60803126, the Zhejiang Province Natural Science Foundation under Grant No. Z1080979.

References

[1] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury, "A Survey on Wireless Multimedia Sensor Networks," *Computer Networks*, vol. 51, pp. 921-960, 2007.

[2] P. Kulkarni, D. Ganesan, P. Shenoy, and Q. Lu, "SensEye: A Multitier Camera Sensor Network," in *ACM Multimedia*, Singapore, 2005.

[3] A. Rowe, D. Goel, and R. Rajkumar, "FireFly Mosaic: A Vision-Enabled Wireless Sensor Networking System," in *IEEE RTSS*, Tucson, Arizona, USA, 2007.

[4] H. Lee, H. Dong, and H. Aghajan, "Robot-Assisted Localization Techniques for Wireless Image Sensor Networks," in *IEEE SECON*, Reston, VA, USA, 2006.

[5] I. Rekleitis and G. Dudek, "Automated Calibration of a Camera Sensor Network," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Saint-Hubert, Que., Canada, 2005.

[6] A. Barton-Sweeney, D. Lymberopoulos, and A. Savvides, "Sensor Localization and Camera Calibration in Distributed Camera Sensor Networks," in *International Conference on Broadband Communications, Networks and Systems*, San Jose, CA, 2006.

[7] H. Lee and H. Aghajan, "Collaborative Node Localization in Surveillance Networks Using Opportunistic Target Observations," in *ACM International Workshop on Video Surveillance and Sensor Networks*, Santa Barbara, California, USA, 2006.

[8] D. Devarajan, R. J. Radke, and H. Chung, "Distributed Metric Calibration of Ad Hoc Camera Networks," *ACM TOSN*, vol. 2, pp. 380-403., 2006.

[9] H. Bidgoli, "*Hacking Techniques in Wireless Networks*," in the Handbook of Information Security.

[10] J. Polastre, R. Szewczyk, and D. Culler, "Telos: Enabling Ultra-Low Power Wireless Research," in *IEEE IPSN*, 2006.

[11] GreenOrbs, http://www.greenorbs.org/

[12] L. Mo, Y. He, Y. Liu, J. Zhao, S. Tang, X.-Y. Li, and G. Dai, "Canopy Closure Estimates with GreenOrbs: Sustainable Sensing in the Forest," in *ACM SenSys*, Berkeley, California, 2009.

[13] R. Mangharam, A. Rowe, R. Rajkumar, and R. Suzuki, "Voice over Sensor Networks," in *IEEE RTSS*, Rio de Janeiro, Brazil, 2006, pp. 291-302.

[14] T. F. Abdelzaher, S. Prabh, and R. Kiran, "On Real-time Capacity Limits of Multihop Wireless Sensor Networks," in *IEEE RTSS*, Lisbon, Portugal, 2004.

[15] T. Yan, D. Ganesan, and R. Manmatha, "Distributed Image Search in Sensor Networks," in *ACM SenSys*, Raleigh, NC, USA, 2008.

[16] P. Kulkarni, D. Ganesan, and P. Shenoy, "The Case for Multi--tier Camera Sensor Networks," in *ACM NOSSDAV*, Washington, USA, 2005, pp. 141 - 146.

[17] W.-C. Feng, E. Kaiser, W. C. Feng, and M. L. Baillif, "Panoptes: Scalable Low-Power Video Sensor Networking Technologies," *ACM TOMCCAP*, vol. 1, pp. 151-167, May 2005.

[18] H. R., S. J., and O.-H. T., "Applying Video Sensor Networks to Nearshore Environment Monitoring," in *IEEE Pervasive Computing*. vol. 2, 2003, pp. 14-21.

[19] S. Kumar, T. H. Lai, and A. Arora, "Barrier Coverage With Wireless Sensors," in *ACM MobiCom*, 2005.

[20] Z. Guo, Y. Guo, F. Hong, X. Yang, Y. He, Y. Feng, and Y. Liu, "Perpendicular Intersection: Locating Wireless Sensors with Mobile Beacon," in *IEEE RTSS*, Barcelona, Spain, 2008.

[21] H.-l. Chang, J.-B. B. Tian, T.-T. Lai, H.-h. Chu, and P. Huang, "Spinning Beacons for Precise Indoor Localization," in *ACM SenSys*, Raleigh, NC, USA, 2008.

[22] Z. Zhong and T. He, "Achieving range-free localization beyond connectivity," in *ACM SenSys*, 2009.

[23] C. Peng, G. Shen, Y. Zhang, Y. Li, and K. Tan, "BeepBeep: A High Accuracy Acoustic Ranging System using COTS Mobile Devices," in *ACM Sensys*, Sydney, Australia, 2007.

[24] O. Faugeras, F. Lustman, and G. Toscani, "Motion and Structure from Motion from Point and Line Matches," in *ICCV*, pp. 25-34,London, 1987, 1987.

[25] X. Liu, P. Kulkarni, P. Shenoy, and D. Ganesan, "Snapshot: A Self-Calibration Protocol for Camera Sensor Networks," in *International Conference on Broadband Communications, Networks and Systems*, San Jose, CA, USA, 2006.