Moving Targets Detection and Localization in Passive Infrared Sensor Networks

Zhiqiang Zhang, Xuebin Gao
Graduate University, Chinese Academy of Science
Zhangzhiq105@mails.gucas.ac.cn

Jit Biswas, Jian Kang Wu
Institute for Infocomm Research, Singapore
jiankangwu@gmail.com

Abstract - This paper presents the method for the detection and localization of moving targets in passive infrared (PIR) sensor networks in both indoor and outdoor settings. It reports our design and implementation of PIR sensor network, especially, we proposed a detection algorithm, which uses Adaptive Threshold with Constant False Alarm Rate; and developed a localization algorithm using direction search in the grid space of the network. The experimental results have shown that our PIR sensor network can detect and locate the moving targets with reasonable accuracy.

Key words: PIR sensor network, detection, localization

1. Introduction

Wireless sensor network can be used to perform a variety of tasks such as environment monitoring, geophysics, surveillance, infrastructure protection, and navigation [1, 2, 3]. In many applications, sensor network is designed to detect, locate and track one or more dynamic targets. Sensor networks are constrained by their limitations of communication, computation and sensing ability of each sensor node. Efficient collaborative signal processing algorithms should be designed to explore the collective behavior of the massively embedded sensor networks.

In PIR sensor network, PIR sensors are very sensitive to noise. Even a little change in the environment such as wind or sunshine may trigger false detection. As such the detection algorithms such as in [4, 5] are hardly applicable here.

So far, there are three types of sensing methods for target localization: time delay of arrival (TDOA), direction of arrival (DOA) and received signal strength or power. In practice, the TDOA techniques require highly accurate measurement or estimation of time delay while the third method requires accurate calibration of sensors. In [3], a target localization algorithm is presented based on acoustic signal emitted by the target. PIR sensors have quite unique sensing model, thus, we have to develop dedicated detection and localization algorithm.

The paper is organized as follows. The prototype system is briefly described in Section 2. Section 3 introduces the PIR sensing model. Section 4 and 5 give detailed treatment on our detection algorithm and localization algorithm, respectively. The experimental results are given in section 6. Section 7 concludes this paper and points out open research issues and future work.

2. Prototype System

2.1 Hardware and Networking

The prototype system has 8 sensor nodes and one base station as shown in Figure 1. Here we use Crossbow's MSP410CA [6, 7] hardware platform as our sensor nodes and mica2 as the base station. MSP410CA consists of an Atmel Atmega128 microcontroller and Chipcon CC1000 radio module. It is equipped with a sensor board that has PIR sensors on it. We use the quad passive infrared detector with IR transparent windows on passive infrared sensors to detect and locate moving targets which emit heat. The PIR sensor as shown in Figure 2 is a set of four sensing elements placed at four sides of the board. They are dual element sensors designed for detecting motion of thermally radiating body.
the sensor to detect an “event”, a warm object must cross from one beam region to the next. This will create two outputs from the sensor. One represents the strength of PIR signal while the other called Quad represents which sensing element has been activated by the target.

2.2 Sensor Scheduling

Although there is MAC layer for MICA 2 to deal with communication collision problem, we found that the packet loss in our experiments is not tolerable: when nodes sample data at a rate of 256ms and then send the collected data, the resulted packet loss is as high as 40-60%. To overcome this problem, a time division sampling and communication scheme is proposed, and shown in Figure 3.

![Figure 3. Time division base schedule strategy](image)

In the scheme, time synchronization is implemented where the server initiates the system by sending a SYNC message containing its own clock. Upon receiving this beaconing message, each sensor adjusts its clock to that of the server [9]. Assuming the network has N sensor nodes, the time is divided into periodic cycles. Within each cycle, a sampling slot is used for all the nodes to sample data while a predefined slot is assigned to each PIR sensor to communicate with the base station. Both in the sensing slot and transportation slot, all the sensors will work properly without mutual inference. Another advantage of the scheme is that it solves the aggregation problem shown in figure 4 by forcing all the sensors performing data sampling at the same time.

![Figure 4. Aggregation problem](image)

Using this time division scheme, when we use 8 nodes and sample rate of 50ms, the packet loss is less than 1%.

3. PIR Sensing Model

The radiation of infrared from the source is dependent on many factors, including surrounding environment, frequency of infrared, temperature of the source, strength of wind, etc.. To simplify the problem, we develop an infrared radiation model in a simplified situation.

According to Stefan-Boltzmann law, since infrared waveform is additive, the infrared wave intensity signature received by each sensor is:

$$x_i(n) = s_i(n) + v_i(n)$$

Where $s_i(n) = \gamma_i\sum_{k=1}^{K} IR_k \exp(-\rho ||\rho - r_i||)$ and $i=1,2,...N$, $v_i(n)$ is zero-mean additive Gaussian white noise, $\gamma_i$ is the gain of the sensor. $\rho$ and $r_i$ are the location of the kth target and sensor respectively. $IR_k$ is the kth target’s infrared intensity. $\varphi$ is infrared radiation attenuation constant.

Assuming $s_i(n)$ and $v_i(n)$ are uncorrelated, $E[v_i(n)] = 0$, $E[s_i(n)]$, and then we get:

$$E[x_i^2(n)] = E[s_i^2(n)] + E[v_i^2(n)]$$

We denote $Y_i(n) = E[x_i^2(n)], S_i(n) = E[s_i^2(n)]$, $e_i(n) = E[v_i^2(n)], S_i'(n) = E[IR_k^2], g_i = \gamma_i^2$, then we get

$$Y_i(n) = S_i(n) + e_i(n)$$

$$= g_i \sum_{k=1}^{K} S_i'(n) \exp(-2\varphi ||\rho - r_i||) + e_i(n)$$

In fact, $e_i(n)$ is now independent $\chi^2$ distribution. For simplicity, we assume it is still an additive Gaussian White noise with mean $u_i(n)$ and variance $\sigma_i^2(n)$.

4. Adaptive Threshold with Constant False Alarm Rate

We use adaptive threshold algorithm with constant false alarm to detect the moving target. The algorithm consists of 3 steps: noise elimination, signal energy calculation, adaptive threshold detection.

For simplicity, in this section, we will omit sensor subscript in equations.

4.1 Noise Elimination

In outdoor environment, the performance of PIR sensor depends heavily on the weather conditions, including wind, temperature and humidity. Wind makes the air move and grass and trees swing, causing the thermal field to change since the air temperature is not uniform and grass and trees have different temperatures [8]. Assuming that the environment changes are slow, and reflected as low frequency components (less than 1 Hz) in the detected signal. This motivates us to design a high
pass filter to filter out the noise components in PIR signal. The high-pass filter has the following form:

$$H(z) = \frac{1 + \alpha}{2} - \frac{1 - z^{-1}}{1 - az^{-1}} \text{ where } |a| < 1$$  \hspace{1cm} (4)$$

Figure 5 shows the frequency responses of the filter when $\alpha$ changes. In our system, the sampling rate is 20Hz.

![Frequency Response Graph](image)

Because of the significant direct current element in the raw signal, general high pass filter will set $m(0) = (1 + \alpha) \cdot y(0) / 2$ and bring in boundary effect. To avoid this effect, we let

$$\begin{align*}
m(0) &= 0; \\
m(k) &= \frac{1 + \alpha}{2} (y(k) - y(k - 1)) + an(m(k - 1))
\end{align*}$$  \hspace{1cm} (5)$$

The experiments show that the filter can increase S/N ratio significantly.

### 4.2 Signal Energy Computing

The energy of the signal corresponds to the thermal field variation caused by the target movement. Assume that the target is 4 meters from the sensor, the angle of beam width is 15°, then the actual beam width is 0.3 meter. Considering that, our sampling rate is 20 samples per second, and a stable movement to cross two regions (0.3 meter) would take about 0.3 second. There, we shall compute the signal energy corresponding to the target movement over 8 samples, as follows:

$$s(k) = \sum_{i=0}^{7} m(k + i)^2$$  \hspace{1cm} (6)$$

### 4.3 Constant False Alarm Rate Detector

Detection by hard decision threshold produces output $v(k)$. The $v(k)$ satisfies equation:

$$v(k) = \begin{cases} 
1 & \text{when } S(k) > Th(k) \\
0 & \text{when } S(k) \leq Th(k)
\end{cases}$$  \hspace{1cm} (8)$$

The threshold must be set based on the noise level. Here, the noise level is changing with the environments. In order to adapt the threshold to the noise level, first, we estimate the mean $u(0)$ and standard deviation $\sigma(0)$ of the noise at the initialization stage, assuming that there is no presence of target and that the noise is independent Gaussian process. We define the threshold as $Th(0) = u(0) + C\sigma(0)$, where $C$ is a constant chosen to yield a desired constant false alarm probability.

The threshold can be updated as follows: if $v(k)=1$, then the threshold keeps unchanged, $Th(k)=Th(k-1)$; otherwise, it is updated as follows:

$$\begin{align*}
u(k) &= \alpha n(k - 1) + (1 - \alpha) S(k) \\
\sigma^2(k) &= \alpha \sigma^2(k - 1) + (1 - \alpha) (S(k) - u(k))^2 \\
Th(k) &= u(k) + C\sigma(k)
\end{align*}$$  \hspace{1cm} (9)$$

Where $\alpha$ is a “memory factor” between 0 and 1.

The probability of False alarm will be:

$$P_{fa} = \frac{1}{\sqrt{2\pi} \sigma(n)} \int_{0}^{\infty} \exp \left( \frac{-(x-u(n))^2}{2\sigma^2(n)} \right) dx$$  \hspace{1cm} (10)$$

Denoting $x' = \frac{x-u(n)}{\sigma(n)}$, then equation (10) can be simplified to:

$$P_{fa} = \frac{1}{\sqrt{2\pi} \sigma(n)} \int_{0}^{\infty} \exp \left( -\frac{x'^2}{2} \right) dx$$  \hspace{1cm} (11)$$

Because $C$ is a constant, then the false alarm will keep as a constant.

The probability of detection will be:

$$P_{d} = \frac{1}{\sqrt{2\pi} \sigma(n)} \int_{0}^{\infty} \exp \left( \frac{-(x-u(n)-S(n))^2}{2\sigma^2(n)} \right) dx$$  \hspace{1cm} (12)$$

$$= \frac{1}{\sqrt{2\pi} \sigma(n)} \int_{0}^{\infty} \exp \left( \frac{1}{2} \left( \frac{x-u(n)}{\sigma(n)} - \frac{S(n)}{\sigma(n)} \right)^2 \right) du$$

$$= \frac{1}{\sqrt{2\pi} \sigma(n)} \exp \left( \frac{1}{2} \left( \frac{u}{\sigma(n)} - \frac{S(n)}{\sigma(n)} \right)^2 \right) du$$

The equation (12) is only dependent on $S/\sigma(n)$, which is the infrared energy signal’s SNR. After the operations described in section 4.1, the noise level is very low, then the SNR is very high which causes the probability of detection to approach one.

In single target situation, the probability of detection will be:

$$P_{d} = \frac{1}{\sqrt{2\pi} \sigma(n)} \int_{0}^{\infty} \exp \left( \frac{1}{2} \left( u - gS \exp \left( -2\rho \|n\| \right) \right)^2 \right) du$$  \hspace{1cm} (13)$$

The right hand side of equation (13) decreases as the distance between sensor and target increases. Then we can conclude that the detection effect will decline as the distance between sensor and target increases.

Considering the continuity of moving and detection, here we define a counter: $D_S$ which is the number of successive positive detections to indicate the existence of the target. It makes the algorithm...
more robust to the short burst errors.

Following we will introduce the probability of false alarm and detection after the count operation. Here we assume that each detection of CFAR detector is independent with each other. Then we get:

$$P'_{fa} = (P_{fa})^{d}$$

(14)

Consider the real world situation, $P_{fa} < 1$ or even $P_{fa} < 0.5$, then $P'_{fa} << P_{fa}$ and $P'_{fa}$ approaches zero and is still a constant.

$$P'_{d} = (P_{d})^{d}$$

(15)

Since infrared energy signal’s SNR is very high, and then $P'_{d}$ will approach one but less than one which results that $P'_{d} < P_{d}$ and $P'_{d}$ also approaches one.

Generally, the count operation reduces the probability of false alarm greatly and also results the probability of detection declines which is still in acceptable range in real sense.

5. Direction Search Localization

After detection of the target, we need to provide the location of the target. This section is dedicated to localization algorithm. In the prototype system, the sensor nodes are placed as shown in Figure 6. The 4 sensing regions of each sensor node is depicted on the right of the figure. Therefore, in figure 6, the sensing area is divided into 19 cells through Quad base line (denoting $i^{th}$ cell by $S_i$). The algorithm is used to decide which cell the target is in.

![Figure 6. sketch map of sensing area](image)

![Figure 7. Quads bit level representation](image)

Figure 7 describes a bit level representation of Quads from 8 nodes. Every node is assigned 4 bits and each bit for its 4 quads respectively. Every bit can be “1” or “0”. “1” represents a positive detection from the corresponding quad and node while “0” represents no detection. Here we denote this structure Node-Quad (NQ).

First, we construct a significant bits table “$T_{sig}$” for the sensing area. Then, $T_{sig} = \{NQ_i | i = 1, 2, \ldots, 19\}$, where $NQ_i$ is a row of the table and only relates to $S_i$. $NQ_i$ is the significant bits of $S_i$ which means there is a target in $S_i$ if and only if $NQ_i$ is satisfied. $S_i$ is the intersection of nodes’ quad while $NQ_i$ indicates intersection can be obtained by the least nodes. For example: $S_2$ is the intersection of Node 1’s Quad 3 and Node 2’s Quad 1, so $NQ_2$ is 0000 0000 0000 0000 0000 0000 0001 0100 or 0X00000014 in hex. Table 1 summarizes $NQ_i$ where $i = 1, 2, \ldots, 19$.

<table>
<thead>
<tr>
<th>Node</th>
<th>NQ</th>
<th>Hex</th>
</tr>
</thead>
<tbody>
<tr>
<td>S1</td>
<td>NQ1</td>
<td>0X0010010</td>
</tr>
<tr>
<td>S2</td>
<td>NQ2</td>
<td>0X0080012</td>
</tr>
<tr>
<td>S3</td>
<td>NQ3</td>
<td>0X0018002</td>
</tr>
<tr>
<td>S4</td>
<td>NQ4</td>
<td>0X0000014</td>
</tr>
<tr>
<td>S5</td>
<td>NQ5</td>
<td>0X0088022</td>
</tr>
<tr>
<td>S6</td>
<td>NQ6</td>
<td>0X0014000</td>
</tr>
<tr>
<td>S7</td>
<td>NQ7</td>
<td>0X0808002</td>
</tr>
<tr>
<td>S8</td>
<td>NQ8</td>
<td>0X0080020</td>
</tr>
<tr>
<td>S9</td>
<td>NQ9</td>
<td>0X0000014</td>
</tr>
<tr>
<td>S10</td>
<td>NQ10</td>
<td>0X0880022</td>
</tr>
<tr>
<td>S11</td>
<td>NQ11</td>
<td>0X0140000</td>
</tr>
<tr>
<td>S12</td>
<td>NQ12</td>
<td>0X0808020</td>
</tr>
<tr>
<td>S13</td>
<td>NQ13</td>
<td>0X0800202</td>
</tr>
<tr>
<td>S14</td>
<td>NQ14</td>
<td>0X0000140</td>
</tr>
<tr>
<td>S15</td>
<td>NQ15</td>
<td>0X8800220</td>
</tr>
<tr>
<td>S16</td>
<td>NQ16</td>
<td>0X1400000</td>
</tr>
<tr>
<td>S17</td>
<td>NQ17</td>
<td>0X0800240</td>
</tr>
<tr>
<td>S18</td>
<td>NQ18</td>
<td>0X8400020</td>
</tr>
<tr>
<td>S19</td>
<td>NQ19</td>
<td>0X0400040</td>
</tr>
</tbody>
</table>

6. Simulation and Experiments

To illustrate the performance of the detector, MATLAB simulation is brought in using collected
real measurement data in a sunny and windy afternoon. Figure 8(a) shows the collected data while Figure 8(b) is the power spectrum of the signal. There is a moving target in the sensing area between 55s and 65s. In Figure 8(b), noise energy and signal energy mix together and can’t be distinguished from each other.

![Figure 8. PIR reading from MSP410](image)

(b) Power Spectrum of PIR data

Figure 9 (a) shows the filtered PIR data while Figure 9(b) shows the power spectrum of filter PIR signal. In the spectrum, noise’s energy was restrained hugely. It’s clear to distinguish noise and signal and determine when a target is moving in and out.

![Figure 9. Filtered PIR Readings](image)

Figure 10 depicts the moving targets detection result using the adaptive threshold detection algorithm. It illustrates that the detection result is consistent with the actual situation. In the figure, the red line represents adaptive threshold. It’s clear that the threshold is adaptive to noise level.

Figure 11(a) and 11(b) are screen shots of the prototype system, showing two experiments of single target and multi-target detection and localization, respectively. The experiments have shown that when a moving target is in a cell of the sensing area, it will be detected and correctly located. The prototype system has been running since August 2006, it demonstrated that the detection and localization algorithms are quite robust to multi-targets.

Figure 12 depicts the respective correct localization ratio and false alarm ratio when we walk in the sensing area ten times. From the figure, we can get that the localization accuracy is about 90% percent and the false alarm is about 0.
Further work is on the development of Bayesian network estimation for the tracking the moving targets.

REFERENCES


7. Conclusion and Future Work

This paper presents a passive Infrared sensor networks for real time moving targets detection. We start by discussion of PIR sensor model, and then highlighted architecture of prototype system based on the sensor model. Detection algorithm using adaptive threshold with constant false alarm rate, and localization algorithm using direction search in the grid of sensing cells are discussed in details. Real-time tests and offline simulations have demonstrated the effectiveness of the algorithms.

However, due to the limitations of sensor hardware, we were not able to obtain the directional information up to the level of further division by Fresnel lens. As such, more accurate localization is still under development.