

## Collaborative Sensing for Heterogeneous Sensor Networks\*

KEJIANG XIAO<sup>1</sup>, RUI WANG<sup>2,+</sup>, CAI CAI<sup>3</sup>, SHAOHUA ZENG<sup>1</sup> AND ZAHID MAHMOOD<sup>4</sup>

<sup>1</sup>*Information and Communication Company of State Grid Hunan Electric Power Company  
Changsha, 410007 China*

<sup>2</sup>*Institute of Computing Technology  
Chinese Academy of Sciences  
Beijing, 100190 China*

<sup>3</sup>*Department of Basic Curriculum  
Changsha Environmental Protection College  
Changsha, 410004 China*

<sup>4</sup>*School of Computer and Communication Engineering  
University of Sciences and Technology Beijing  
Beijing, 100083 China*

*E-mail: 985089629@qq.com; wangrui@ustb.edu.cn*

Collaboration between low-quality sensor and high-quality sensor can achieve tradeoff between accuracy and energy efficiency in heterogeneous sensor networks (HSNs). Generally, HSNs are deeply integrated with dynamic physical environments. The monitored target's dynamics are the most important and common factors of the dynamic environments. Some important parameters (such as active opportunity, sampling frequency and sampling time) fail to adapt to the changes, which undermines the collaboration's performance when the state of the monitored target changes. Even the system performance is not up to user requirements or large amounts of energy are consumed. To solve this problem, we propose an adaptive collaboration scheme named EasiAC by the collaboration between magnetic and camera sensors. First, for the dynamics of the monitored target, EasiAC utilizes the magnetic sensors to predict the target's state via Bayesian filtering based method. Second, to achieve good performance of such collaboration, EasiAC adjusts the camera sensors' active opportunity, optimal sampling frequency and sampling time dynamically according to the estimated results from the magnetic sensors. Finally, a boosting based algorithm named BbTC is proposed to make classification for the target to achieve high accuracy (average classification accuracy is more than 98%). We evaluate EasiAC method through extensive simulations and real road environment experiments. The results demonstrate that EasiAC needs less energy consumption (saving 97% energy) than traditional solutions, while maintaining the performance at acceptable level (average image integration ratio is 90%) in the presence of target's dynamics.

**Keywords:** Bayesian filtering, dynamic environments, collaboration, sampling frequency, target classification, wireless sensor networks.

### 1. INTRODUCTION

Heterogeneous sensor networks (HSNs) which can support multiple applications [1] for HSNs with multiple types of sensors are gaining popularity in diverse fields. In order to measure the same physical quantity, different type's sensor is characterized by its own performance and energy efficiency. In most cases, low-quality sensors such as magnetic

---

Received March 21, 2015; revised November 3, 2015; accepted December 21, 2015.

Communicated by Jiann-Liang Chen.

\* This paper is supported by National Natural Science Foundation of China (NSFC) under Grant No. 61379134.

+ Corresponding author: Rui Wang (wangrui@ustb.edu.cn)

sensor are energy-efficient, but provide a very limited resolution. On the other hand, high-quality sensors such as camera sensor can provide more accurate characterization of sensed phenomenon at the cost of higher energy usage. Therefore, collaboration between the low-quality sensor and high-quality sensor can achieve tradeoff between accuracy and energy efficiency. In real applications, low-quality sensors can provide a coarse-grained characterization of the sensing field or trigger an event [2]. Then, accurate, but power-limited, sensors can be activated with measurements which are used to improve the coarser description or complete complex tasks. Therefore, Accuracy can be traded off with energy efficiency through collaboration among low-quality sensors and high-quality sensors.

Normally wireless sensor networks are deeply integrated with physical environments. The uncertainties of various physical environments are ubiquitous. These uncertainties include the dynamics of monitored target, stochastic sensor noises and unpredictable environment changes. But most of existing collaboration methods use fixed parameter configuration and processing strategies, which fail to adapt to various physical uncertainties. Thus the performance of collaboration in HSNs is inevitably undermined by these uncertainties, which pose great challenges to the collaboration between low-quality sensors and high-quality sensors.

Next, we give an example to illustrate the problem mentioned above. The collaboration between magnetic sensor and camera sensor is utilized to make classification for targets in this example. Magnetic sensor is used to detect the arrival of targets, and trigger the camera sensor node to classify the targets when they are detected. However, if the distance between the targets and camera sensor node is too far or the targets have not yet entered into the camera's field of vision (FOV), the targets will not be sensed. Only the targets which are in the camera sensor's sensing range can be detected by the camera sensor. If the magnetic sensor nodes detect the target, the target is still outside the camera sensor's sensing range or the sampling frequency of the camera sensor is too high and the sampling time is too long, the camera sensor node will gather a large amount of useless images if the camera is triggered at this moment. Thus it causes some unnecessary overheads. But the precise time when the target enters into the camera's FOV, the camera sensor's optimal sampling frequency and sampling time cannot be obtained directly. Therefore, we cannot obtain the camera's active opportunity, sampling frequency and sampling time precisely. In this paper, active opportunity stands for the time when the target enters into the camera's sensing range, when the target is detected by the magnetic sensors in the vicinity of the camera sensor.

Thus, in order to obtain the camera sensor's active opportunity, optimal sampling frequency and sampling time precisely, we need to obtain the target state in real time. As the active opportunity arrives, the camera sensor node is triggered with the optimal sampling frequency and sampling time, which can reduce the quantity of the redundant images and maintain the performance at acceptable level. The active opportunity, sampling frequency and sampling time of the high-quality sensors are key issues for energy-efficiency and performance in the process of collaboration between low-quality sensors and high-quality sensors. In this paper, we utilize the collaboration between the magnetic sensor and camera sensor to achieve the trade-off between performance and energy efficiency. We note that the magnetic sensors and camera sensors can be easily extended to other low-quality sensors and high quality sensors.

The major challenges in this paper include two aspects: 1) how to design a light-weight filtering algorithm to predict the target's state for the sensor node's limited resources; 2) how to compute the active opportunity, sampling frequency and sampling time of the camera sensor dynamically. For the two challenges, we propose an adaptive collaboration (EasiAC) method, which uses magnetic sensor nodes to predict the state of the target via distributed Bayesian filtering. Then EasiAC tunes the camera sensor's active opportunity, sampling frequency and sampling time dynamically according to the estimated results from the magnetic sensors. Thus EasiAC can decrease the camera's active time, sampling frequency and sampling time compared with the traditional method that when the low-quality sensor detects the target, the camera is triggered immediately and works with the maximum sampling frequency. Finally, because boosting algorithm [3] can combine weak classifiers to form strong classifier to get high performance and low complexity, a boosting based algorithm is proposed to make classification for camera sensors. This paper is the extension of our previous work [4]. Difference from [4], we compute optimal sampling time according to target state and propose a boosting based classification method in dynamic environments in this paper. The contributions of this paper are summarized as follows.

- (1) We provide general principles guiding the prediction of the dynamic target's state precisely in HSNs. The magnetic sensor nodes are used to predict target's state via distributed Bayesian filtering and overcome the limitations of the sensors' resources to get the target's state.
- (2) We derive active opportunity, sampling frequency and sampling time of the camera sensor node according to the magnetic sensor nodes' estimated results to deal with the problems of the dynamics of the monitored target. They can adapt the changes of the monitored target and achieve tradeoff between performance and energy-efficiency.
- (3) We propose a boosting based algorithm named BbTC algorithm. BbTC can achieve high performance and low complexity when it is utilized to make classification for the dynamic target.

The rest of the paper is organized as follows. Section 2 reviews related work. Section 3 introduces our collaborative sensing scheme. The target classification will be discussed in section 4. Section 5 presents the simulations experiments and real road experiments. Section 6 concludes this paper.

## 2. RELATED WORK

There are some works to optimize system performance based on the dynamic changes in the network environments, such as an energy-saving architecture BiSNET [5] which proposes a distributed self-management of the sensor network architecture. It can adjust sleep time and the network response time dynamically according to the dynamic changes of network environment, and balances each node's sensing cycle and energy consumption adaptively. However, BiSNET is mainly used to cope with the issue of dynamic adaptation for single sensor node and there are no collaborations between the sensor nodes. Besides, in [6], the authors present an automatic Markov Decision-based

method to describe the optimal sensor node operation, and select appropriate parameter values to meet application requirements and adapts to environmental change's stimuli. But this method is also only used to adjust single sensor node and does not consider utilizing the collaboration between the sensor nodes to optimize the global system.

As for adaptive sampling method, [7] proposes an adaptive sampling algorithm to dynamically estimate optimal sampling frequency for minimizing energy consumption of a snow sensor via temporal correlation. However, it only can be used in the cases where the process to be monitored exhibits slow variation over time (*e.g.*, monitoring snow composition in mountain slopes for avalanche forecasting). When the monitored phenomena are changed dynamically, it fails to work well. To overcome the uncertainties in camera sensor networks, [8] proposes a robust adaptive synchronization scheme to predict the camera's optimal sampling frequency by exploiting the broadcast nature of wireless communication. But it adjusts the sampling frequency dynamically to monitor the static phenomenon and it fails for the dynamic target. Considering the dynamic property of WSNs, [9] proposed two adaptive algorithms: one is for adapting the sample with varying of precision requirement, and the other is for adapting the sample rate with varying of sensed data. But it does not consider the dynamics problem.

Most of the existing works [10-12] do not considering the active opportunity of high-quality sensor node, and the high-quality sensor node is triggered immediately when the target is detected by low-quality sensors. For example, [10] proposes an adaptive system-level calibration method. Specifically, the method uses data fusion results of low-precision sensors to judge whether target appears. Once the target was detected, camera sensor node will be triggered immediately. But it doesn't consider the problem of camera sensor's active opportunity. [11] proposes fidelity-aware utilization controller method for CPS (Cyber-physical System) monitoring system in the environment without prior arrangements combined with low-quality sensors and camera sensors for large-scale ad-hoc monitoring. The use of low-quality sensors is to detect the target, and then camera sensor node is triggered to achieve the control objectives of camera workload by optimizing fusion threshold and adjusting CPU utility. But it also doesn't consider the impact of the camera's active opportunity on system performance and overhead when the state of target is highly dynamics.

As for target classification, literature [5] has studied classification problem extensively which also has attracted considerable attention in the area of wireless sensor networks (WSNs). Recently, more and more researchers have realized that the target classification plays an important role in surveillance applications of WSNs. Thus, some works [13-16] began to focus on the target classification. For example, target classification result is achieved by a static classifier in a centralized manner in [13, 14]. The classifier learning is based on a supervised learning method, and localization and classifier learning are both executed by the progressive data-fusion paradigm, with specifically designed sensor-node-selection strategy [17]. The authors of [18] modeled the spatial-temporal signal field generated by an object as a band-limited stationary argotic Gaussian field. It analyzed the classifier performance for both soft and hard decision fusion across regions assuming noise-free as well as noisy communication links between nodes. However, these works often do not consider the different of the local classifiers' weight in dynamic environments, and usually can not achieve high performance, when the fusion node fuses the decisions from the local sensor nodes.

There are also a few works about target classification based on boosting method [19-21]. For example, [19] propose a novel and fast weighting method using a boosting algorithm to find the sensor area contributing to accurate discrimination of vowels. A boosting algorithm driven distributed template matching approach is present in [20]. The boosting method is employed to enhance accuracy by activating only a subset of sensors optimized in terms of power consumption and can achieve a given bound accuracy criterion. However, they all do not utilize boosting algorithm for camera sensors.

### 3. COLLABORATIVE SENSING SCHEME

In this section, we first make an overview of our collaborative sensing scheme EasiAC. And then we introduce the design and implementation of EasiAC. Finally, EasiAC algorithm will be analyzed.

#### 3.1 An Overview of EasiAC

As shown in Fig. 1, the collaboration between magnetic sensor and camera sensor is utilized to make classification for targets. Magnetic sensor is used to detect the arrival of targets, and trigger the camera sensor node to classify the targets when they are detected. Specially, EasiAC includes three parts: magnetic sensors, camera sensors and base station. Target can be sensed by magnetic sensor nodes, and a Bayesian filtering based approach is employed to iteratively predict the target's state. Then, active opportunity, optimal sampling frequency and sampling time of the camera sensors is computed according to the target's state predicted by Bayesian filtering. When the active opportunity occurs, the camera sensor is triggered and works with the optimal sampling frequency. After the sampling time, camera sensor node stops sensing and send the images to the Base station. Base station receives the images captured by the camera sensors and process these images to make target classification or tracking. There are already a lot of image processing methods [11, 12], thus we don't consider this part and the target classification will be discussed in section 4.

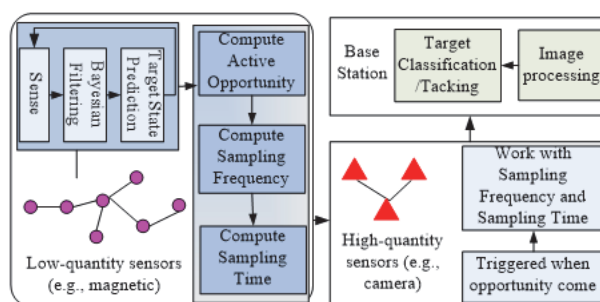


Fig. 1. Modules and interactions in EasiAC.

#### 3.2 Design and Implementation of EasiAC

As the target position changes dynamically, the state of the target is difficult to pre-

dict. Dynamics of the monitored target undermines the collaboration's performance and the system requirements for sensing accuracy may not be satisfied. Even the system requirement can be met, and it may cause some unnecessary energy consumption. To solve this problem, two issues should be considered: (1) how to use the lightweight algorithm to estimate and predict the target's state accurately; (2) how to calculate active opportunity, optimal sampling frequency and sampling time, making them self-adaptive to the changes of the target's state and adjust dynamically.

For the first issue: Because the target changes dynamically, predicted results of extrapolation method [22] is ineffective. Particle filter [23] can be used to estimate the target's state effectively and get more precise results than extrapolation method, but it will cause a large amount of storage and computation overhead and does not apply to the sensor networks. Bayesian filtering [24] is a recursive algorithm, and it only needs to save the data of a moment before during the operation of the filtering. Because of this feature, Bayesian filtering method is regarded as a lightweight algorithm and its storage and computational overhead are smaller compared with other filtering methods (*e.g.*, particle filter). Besides, it also achieves better prediction. So we consider using Bayesian filtering to predict the moving target's state.

For the second issue: On the one hand, to get the complete features of the target, we should increase the camera sensor's sampling frequency and sampling time, and decrease the active opportunity as much as possible. On the other hand, to decrease the energy consumption, we should increase the active opportunity and decrease the sampling frequency and sampling time as much as possible. Thus, camera sensor's sampling frequency, sampling time and active opportunity should adapt to the target state's changes to reduce energy consumption while maintaining an acceptable performance on the acquired data. Because different targets' velocity is different when they pass through the camera sensor's FOV, we can adjust camera sensor's sampling frequency, sampling time and active opportunity dynamically according to the target's speed.

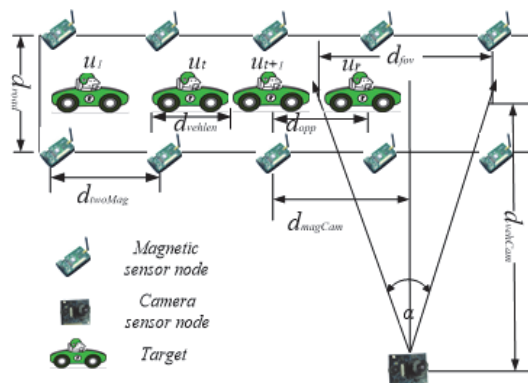


Fig. 2. Adaptive collaboration schematic diagram.

Thus we utilize low-power magnetic sensor nodes to estimate and predict the target's velocity via distributed Bayesian filtering based method, and then compute camera sensor's active opportunity, sampling frequency and sampling time. Next, we derive the

active opportunity, sampling frequency and sampling time of the camera sensor according to general principles that low-quality sensors are used to predict and estimate the target state and trigger the high-quality sensor to work via the estimated results.

### 3.2.1 Active opportunity

As shown in Fig. 2, there are large amount of magnetic sensor nodes and some camera sensor nodes are deployed on both sides of the road. Magnetic sensor is used to detect the arrival of targets, and trigger the camera sensor node to classify the targets when they are detected. We introduce distributed Bayesian filtering to estimate the target location  $p(u_{t+1} | \bar{z}_{t+1})$  from the measurement history  $\bar{z}_t$  in target tracking. The key issue is how to compute the current *a posteriori* distribution  $p(u_t | \bar{z}_t)$  called belief efficiently. To minimize computation and power consumption, we use a leader-based tracker. The leader obtains a belief state  $p(u_t | \bar{z}_t)$  from previous leader at the time  $t$ , and takes a new measurement  $z_{t+1}$ . Through the sequential Bayesian filtering, leader computes the new belief  $p(u_{t+1} | \bar{z}_{t+1})$ , based on the new measurement  $z_{t+1}$  [24]:

$$p(u_{t+1} | \bar{z}_{t+1}) \propto p(z_{t+1} | u_{t+1}) \cdot \int p(u_{t+1} | u_t) \cdot p(u_t | \bar{z}_t) du_t \quad (1)$$

Where  $p(z_{t+1} | u_t)$  is observation's likelihood given the target's location  $u_{t+1}$ ;  $p(u_{t+1} | u_t)$  is related to target dynamics;  $p(u_t | \bar{z}_t)$  is the belief inherited from the previous step.

The initial target location  $u_1$  is known as shown in Fig. 2, the estimated target location  $u_t$  and  $u_{t+1}$  at time  $t$  and  $t+1$ , is calculated as follows:

$$u_t = \sum_{i=1}^k z_i \cdot p(u_t | z_{1:t}), \quad (2)$$

$$u_{t+1} = \sum_{i=1}^k z_{t+1} \cdot p(u_{t+1} | z_{1:t+1}). \quad (3)$$

Where  $k$  is number of grids representing for probability distributions. Therefore, the velocity of the target at time  $t+1$  is  $v_{t+1}$ , which is formulated as follows:

$$v_{opp} = v_{t+1} = u_{t+1} - u_t. \quad (4)$$

Therefore, the time when the target enters into the camera's sensing range is the camera's active opportunity  $t_{opp}$  which is formulated as follows:

$$t_{opp} = d_{opp} / v_{opp} \quad (5)$$

Where  $d_{opp}$  is computed as follows:

$$d_{opp} = d_{magCam} - d_{vehCam} \cdot \tan(\alpha/2) + d_{vehlen}/2. \quad (6)$$

Where  $\alpha$  is the camera's field of vision (FOV).

### 3.2.2 Sampling frequency

To perfectly reconstruct sampled signal, when sensor's sensing environment changes, the sensor's minimum sampling frequency should be adjusted according to Theorem 1.

**Theorem1:** If the perception environments of the sensor are dynamics, the sensor's minimum sampling frequency which can perfectly reconstruct the sampled signal will change dynamically.

**Proof:** Spectrum of signal  $f(t) = (1/2\pi) \int_{-2\pi F_{\max}}^{2\pi F_{\max}} F(\varpi) \exp(-j\varpi t) d\varpi$  is  $F(\varpi) = \sum_{n=-\infty}^{\infty} (1/2F_{\max}) \times f(n/2F_{\max}) \exp(j\varpi n/2F_{\max})$  on the fundamental period  $-2\pi F_{\max} < \varpi < 2\pi F_{\max}$ . When the sensor's sensing environments change, the period of  $\varpi$  and the maximum frequency  $F_{\max}$  in the power spectrum of the signal will change [7]. According to Shannon sampling theorem [25], the minimum sampling frequency is  $2F_{\max}$  which can perfectly reconstruct the signal. Therefore, the sensor's minimum sampling frequency changes dynamically when the sensor is in dynamic environments.

Because of the camera sensor node's limited computing resources, storage resources and system requirements for real-time, the sampling frequency cannot be too high. To capture the complete information of the moving target, sampling frequency cannot be too low and should change dynamically for the dynamics of the monitored target according to Theorem 1. What's worse, the maximum frequency  $F_{\max}$  is generally not a priori available. Fortunately, we found that target speed can be used to determine the optimal sampling frequency through the observation of real experimental data on the road.

The width of the camera sensor's FOV is  $d_{fov}$  as shown in Fig. 2, which is calculated as follows:

$$d_{fov} = 2d_{vehCam} \cdot \tan(\alpha/2). \quad (7)$$

Where  $\alpha$  is the range of the camera sensor's FOV, which is relatively small, so we think that target does uniform straight line motion in the camera sensor's FOV. So the velocity of the target  $v_f$  in the camera's FOV can be calculated as  $v_f \approx v_{opp}$ . Thus, the optimal sampling frequency is calculated as follows:

$$f_{opt} = (\lambda + 2) \cdot v_f / (d_{fov} - d_{vehlen}). \quad (8)$$

Where  $\lambda$  is the number of intact images needed to be captured, which can be adjusted according to the needs of the task. The constant 2 is a noise constant, represents the number of incomplete images that can be tolerated.

### 3.2.3 Sampling time

To obtain integrated images, the camera sensor node needs to have enough sampling time. However, in order to save energy usage, the sampling time of camera sensor node should be adapted to the vehicle changes. Based on the analysis of the real experiments, the optimal sampling time can be computed as follows.

$$T_{opt} = (d_{fov} - d_{vehlen}) / v_f \quad (9)$$

We can get  $\lambda$  number of intact images if we tune the camera sensor's sampling frequency and sampling time adaptively according to Eqs. (8) and (9). Thus camera sensor's sampling frequency and sampling time can adapt to the dynamics of the monitored target to reduce the camera sensors overhead while meeting the user requirements.



Based on the discussion mentioned above, the camera sensor's active opportunity, sampling frequency and sampling time can be calculated according to Eqs. (5), (8) and (9). When the active opportunity  $t_{opp}$  occurs, the camera sensor node is triggered and works with the sampling frequency  $f_{opt}$ . When the target begins to leave the camera sensor's FOV, the camera sensor node stops sampling and goes to sleep. The camera sensor will be triggered again for the next target arrival.

### 3.3 Adaptive Collaboration Algorithm

The pseudo code of our adaptive collaborative algorithm is described in Algorithm 1. In this algorithm, we firstly utilize the magnetic sensor nodes to predict the velocity of the target via distributed Bayesian filtering. Then, we compute the active opportunity, sampling frequency and sampling time of the camera sensor node through the target's velocity. When the active opportunity comes, the camera sensor is triggered and works with the sampling frequency. After the sampling time, the camera sensor stops sensing and the captured images are sent to base station and processed.

The overhead of distributed Bayesian estimation for computation, bits to be communicated and the wireless communication power are  $O(|\kappa| \cdot |belief|)$ ,  $O(|belief|)$  and  $O(|belief| \cdot \|\mathcal{G}_{nextleader} - \mathcal{G}_{leader}\|^{\alpha})$  respectively [24]. Because the active opportunity, sampling frequency and sampling time are computed via the Bayesian filtering, its computation overhead can be ignored. We use X-MAC protocol [26] for the camera sensor node, and the command message is only 6 bytes [27], so the communication cost for triggering camera sensor node can also be ignored. The main cost of the algorithm comes from the camera's sensing and communication [28] (send the images to base station) compared with distributed Bayesian filtering. Thus total energy consumption for *EasiAC* algorithm can be approximated as energy consumption of the camera sensor node's sensing and communication.

---

#### Algorithm 1: Adaptive Collaboration Algorithm

---

$u_t$ : Target position at time  $t$ ;  $v_{max}$ : upper bound on target speed;  $z_t$ : sensor measurement at time  $t$ ;  $Y$ : neighbor list.

```

1   for  $t = 1 \rightarrow n$ 
2       sleep until receive handoff package ( $t, p(u_t | \bar{z}_t)$ )
3       diffuse belief using target dynamics  $p(u_{t+1} | \bar{z}_t) = \int p(u_{t+1} | u_t) \cdot p(u_t | \bar{z}_t) du_t$ 
4       do sensing;
           compute  $z_{t+1}$  and  $p(z_{t+1} | u_{t+1})$ ; compute  $p(u_{t+1} | \bar{z}_{t+1}) \propto p(z_{t+1} | u_{t+1}) \cdot p(u_{t+1} | \bar{z}_t)$ 
6       for sensor  $k \in Y$ , select next sensor node then
7           select  $k_{next} = \arg \max I(u_{t+2}; z_{t+2} | \bar{z}_{t+1})$ 
8       end for
9           Handoff ( $t+1, p(u_{t+1} | \bar{z}_{t+1})$ ) to  $k_{next}$ 
10      end for
11      compute  $u_n = \sum z_n \cdot p(u_n | z_{1:n})$  and  $u_{n-1} = \sum z_{n-1} \cdot p(u_{n-1} | z_{1:n-1})$ 
           compute  $v_{opp} = u_n - u_{n-1}$  and active opportunity  $t_{opp} = d_{opp} / v_{opp}$ 
           compute sample frequency  $f_{opt} = (\lambda + 2) \times v_f / (d_{fov} - d_{vehlen})$ 
           compute sampling time  $T_{opt} = (d_{fov} - d_{vehlen}) / v_f$ 

```

```

12 if active opportunity  $t_{opp}$  occurs then
13     the camera sensor is triggered work with the  $f_{opt}$  and sends the images to the
        base station for processing
14     if the target begins to leave the camera sensor's FOV
15         the camera sensor stops sampling and goes to sleep
16     end if
17 else
18     go back to 12
19 end if

```

---

## 4. TARGET CLASSIFICATION

In this section, we introduce target classification using a boosting algorithm based method and image samples captured by camera sensors. Next, we first discuss feature extraction and feature selection, and then we introduce BbTC algorithm inspired by boosting algorithm.

### 4.1 Feature Extraction and Selection

In order to make classification, we should extract feature from the integrated images captured by camera sensors through some basic image process methods. Note that we take bicycle and car as example for the target to illustrate our scheme in this paper.

#### 4.1.1 Feature extraction

In this paper, to reduce the computation overhead, we first make mean compression for the original image. Then we make background subtraction, first-order gradient, and thresholds binarization for the compressed image respectively. The processed images of minibus, car and bicycle are shown in Fig. 3. To extract the target feature, we obtain the vehicle outline as shown in Fig. 4. The features including the length, width, area, perimeters, length/width (LHR), area/perimeter (PAR) of the vehicle are computed by counting the number of pixel in the vehicle outline according to features extraction algorithm in [29].



Fig. 3. Image process.



Fig. 4. Vehicle outline.

#### 4.1.2 Feature selection

We manually collected sample images, including 259 bicycles, 140 cars. According to the analysis of features of the integrated images, we show that  $LHR$  and  $PAR$  are more helpful features in classification as shown in Fig. 5. Therefore, they are used to classify the vehicles in this paper.

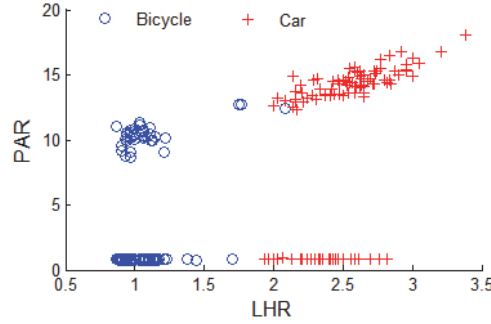


Fig. 5. Feature selection.

#### 4.2 BbTC Algorithm

Boosting algorithm can combine weak classifiers to form strong classifier and weak classifiers can run on cheap off-the-shelf nodes with lower performance. By collaborating sensors wisely, boosting algorithm can achieve high performance and low complexity [3, 20]. Thus, boosting algorithm is suitable to wireless sensor networks and a boosting based algorithm named *BbTC* Algorithm is proposed in this paper. In this algorithm, let  $N$  sensor nodes to classify the monitored target (*e.g.*, vehicles). Each training sample consists of one or many dimension and each dimension is associated with one sensor's data. Local classifier is trained via the collaboration between the data coming from different sensors and the local classifiers with the lowest error are selected during each iteration as follows.

$$\varepsilon_t = \min_s \varepsilon_{t,s} \quad (10)$$

$$h_t(d) = \arg \min_{h_{t,s}} \varepsilon_{t,s} \quad (11)$$

Where  $\varepsilon_{t,s}$  is the classification error of each sensor's local classifier defined by  $\varepsilon_{t,s} = \sum_i w_i |(h_{t,s}(d_i) - y_i)|$ .  $h_t(d)$  is the local classifiers with the lowest error during the  $t$ th iteration.  $h_{t,s}$  means the classifier trained by data from sensor  $s$  during the  $t$ th iteration.

Then the weights of the training samples are updated according to the classification error rate. The weight is defined as follows during iteration  $t+1$ .

$$w_{t+1,i} = w_{t,i} \beta_t^{1-\varepsilon_i} \quad (12)$$

Where  $w_{t,i}$  is the weight of sample  $i$  during the  $t$ th iteration,  $\beta$  can be computed by  $\beta_t = \varepsilon_t / (1 - \varepsilon_t)$ .

In the process, we construct the correspondence between local classifiers and sensors. The corresponding classifiers are combined according to each classifier's weight  $\alpha_t$  to form the virtual strong sensor to improve the system performance in a collaborative way. The final strong classifier can be computed as follows:

$$h_f(d) = \text{sign}\left(\sum_{t=1}^T (1/2)\alpha_t h_t(d)\right) \quad (13)$$

Where  $T$  is the number of iterations.

According to Eq. (3) *BbTC* assigns corresponding weights to the different local classifier according to classification accuracy of local classifiers, and considers complementation of the local classifiers. However, some traditional methods does not consider the impact degree of each local classifier and assigning the same weight to all classifiers. Some other methods only utilize the local classifier with the great weight to make the final decision, and it does not consider the other local classifier with little big contribution to the classification. The most related work is proposed in [16], assign the weight to local classifiers based on the distance between the sensor and the target. However, they do not consider complementary information fully and do not assign the appropriate weight for local classifiers according to their capabilities.

## 5. EXPERIMENTAL EVALUATION

### 5.1 Simulation Experiments

In this section, we conduct the simulation experiments thoroughly in Matlab. The critical collaboration parameters such as active opportunity, sampling frequency and sampling time are evaluated in dynamic environments. Besides, the corresponding image integration ratio, data size and energy consumption are compared with the traditional methods. As for the classification accuracy will be evaluated in the real road experiments for target features can not be obtained in the simulation experiments.

#### 5.1.1 Simulation methodology and settings

Generally, the range of the camera's FOV  $\alpha$  is  $25^\circ$ , the length of the car and bicycle are 4.5m and 1.5m respectively, the velocity range of the car and bicycle are 5.56~11.1 m/s and 2.78~5.83m/s respectively. Based on our experience, we set the sampling frequency  $f_c$  of the magnetic sensor to be 100Hz, and the distance  $d_{magCam}$  between the camera sensor node and the magnetic sensor node to be 16m. According to Eq. (5),  $d_{opp}$  of car and bicycle are 13.85m and 12.35m respectively. Other parameters are shown in Table 1.

In our simulation experiments, 62 magnetic sensor nodes and 1 camera sensor nodes are deployed on both sides of the road; parameter settings and simulation scenario are shown in Table 1 and Fig. 2 respectively. Active opportunity error, sampling frequency and sampling time are analyzed in the simulations. Active opportunity error is the difference between estimated active opportunity and real active opportunity. To evaluate the system performance, we define metrics as follows. *Image integration ratio (IIR)*:  $IIR = n_c/N$ .  $n_c$  is the total integrated image samples;  $N$  is the total image samples. An image sample means the images captured by the camera sensor nodes as it is triggered one time.

**Table 1. Simulation parameter settings.**

Parameter	Value
Area size	300*40 m <sup>2</sup>
Number of magnetic sensor nodes	62
Number of camera sensor nodes	1
Distance between magnetic sensors at the same side $dtwoMag$	10 m
Width of the camera sensor's view $dfov$	8.8 m
Distance between target and camera sensor node $dvehCam$	20 m
Noise of magnetic sensor	0.5
Distance of the road $droad$	7.5 m
Maximum frequency of camera sensor $fmax$	30 Hz
Iterations in Bayesian filtering	25

Integrated image stands for the target completely in the image. Only obtaining integrated image can get the complete feature information of the target. So IIR is an important metric of the performance. *Data size (DS)*: DS is defined as the amount of the data generated by the camera sensor nodes. The data needed to storage and process can be measured by DS, thus it is an important metric of the performance. *Energy consumption (EC)*:  $EC = EC_{sense} + EC_{comm}$ ,  $EC_{sense}$  and  $EC_{comm}$  are energy consumption of the camera's sensing and communication respectively.  $EC_{sense} = t_s \times p_s$ ,  $EC_{comm} = t_c \times p_c$ ,  $t_s$  and  $t_c$  are the time of camera's sensing and communication respectively,  $p_s$  and  $p_c$  are the power of the sensing and communication respectively.  $p_s$  is 238.05mW and  $p_c$  is 279.34mW according to [28].

Besides, to verify the effectiveness of EasiAC, we use three comparison methods, named: ITC, ITImC and RAC method respectively. *ITC method* [11]: when magnetic sensor nodes detect the target, camera sensor node is triggered immediately and works with the maximum sampling frequency. When the target begins to leave the camera sensor's sensing range, the camera sensor stops working and goes into sleep state. *ITImC method*: it is the improved method of ITC method. Its sampling frequency is the same as EasiAC and other aspects are the same as ITC method. *RAC method*: the velocity is select randomly in the target velocity range, and other aspects are the same as EasiAC.

### 5.1.2 Simulation results

In the first set of simulations, we evaluate performance of EasiAC compared with RAC, ITImC and ITC method under different velocity of bicycle and car. Then the performance of EasiAC compared with RAC, ITImC and ITC is evaluated under constant acceleration. Each result is the average of 100 times experiments.

As shown in Fig. 6, EasiAC can adjust active opportunity sampling frequency and sampling time adaptively according to the velocity of the car. EasiAC can get about 100% image integration ratio and has lower energy consumption than ITC and ITImC. Because the velocity is select randomly in target velocity range, active opportunity error of RAC is bigger than EasiAC, and is relatively small in the vicinity of the average velocity, close to EasiAC. But when the velocity is higher or lower, RAC's active opportunity error is relatively big. What is more, sampling frequency and sampling time of RAC does not adapt to the changes of the car's velocity, and is in a relatively stable state. Although RAC method's energy consumption is the same as EasiAC, its image integra-

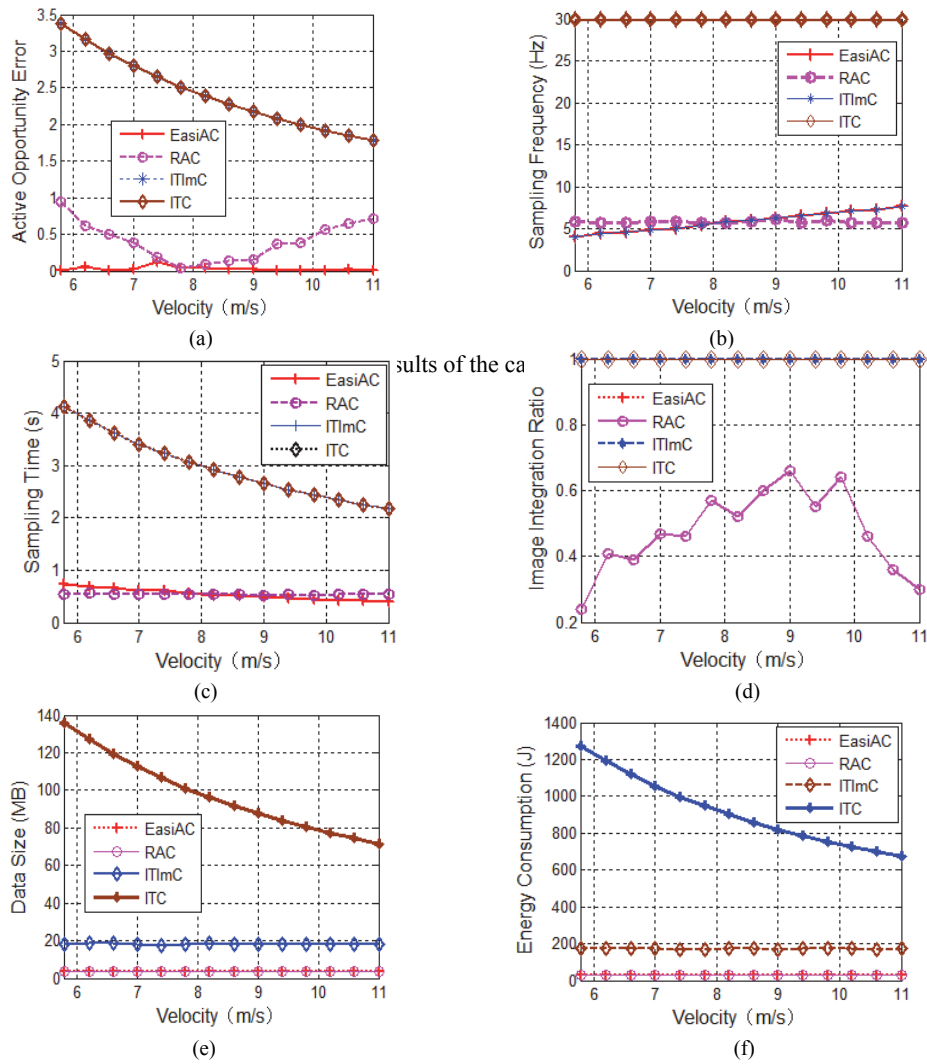


Fig. 6. The comparison results of the car under different velocity.

tion ratio is not more than 70%, and the highest point is in the vicinity of the average velocity for its velocity is selected randomly; Active opportunity error of ITC and ITImC is bigger than EasiAC and RAC, and decreases as the speed increases. ITC's image integration ratio is close to EasiAC, but its energy consumption is higher than EasiAC, because when the car is detected, camera sensor node will be triggered immediately and works with the maximum sampling frequency. ITImC's sampling time is the same as ITC, and the sampling time decreases with the increase of the velocity. When the car is not within the camera sensor's FOV, it has already begun to sample. The camera sensor's sampling time is long enough to meet the needs of image integration ratio, resulting in a large amount of data. Thus ITC method causes large amounts of data and energy consumption coming from the perception, processing and communication. ITImC's im-

age integration ratio is the same as EasiAC and ITC, but its energy consumption is smaller than ITC and bigger than EasiAC and RAC. Because RAC's sampling frequency has self-adaptive like EasiAC and ITImC's active opportunity is the same as ITC.

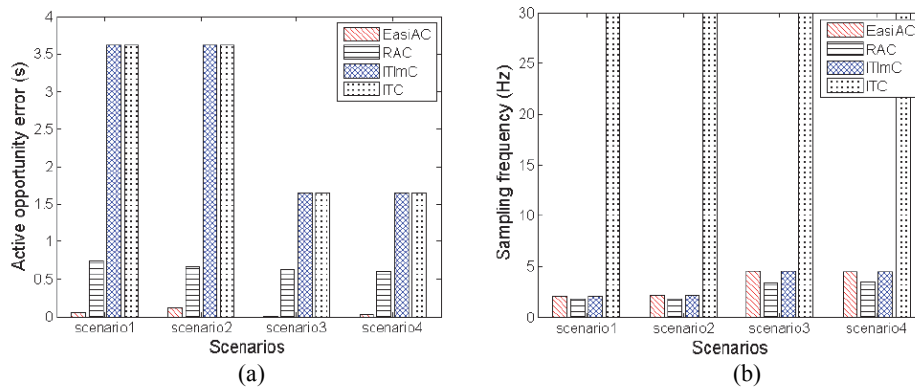
Besides, EasiAC, RAC, ITImC and ITC's comparison results of active opportunity error, sampling frequency, sampling time, image integration ratio, data size and energy consumption under different velocity of the bicycle, which is similar to the results of the car under different velocity. The difference is that the former's active opportunity error is smaller, sampling frequency is higher, and sampling time is higher than the latter, because the car is faster than bicycle.

How does it come out when target's velocity is variable? We set four scenarios as shown in Table 2: (1) bicycle with constant velocity; (2) bicycle with constant acceleration; (3) car with constant velocity; (4) car with constant acceleration. The four methods' comparison results of active opportunity error, sampling frequency, sampling time, image integration ratio, data size and energy consumption under the four scenarios.

**Table 2. Scenarios parameter settings.**

bicycle			car		
scenario1	scenario2		scenario3	scenario4	
constant velocity (m/s)	constant acceleration Initial velocity	acceleration (m/s <sup>2</sup> )	constant velocity (m/s)	constant acceleration Initial velocity	Acceleration (m/s <sup>2</sup> )
5.5	2.5	0.1	11	5	0.2

As shown in Fig. 7, RAC, ITImC and ITC's results of active opportunity error, sampling frequency, sampling time image integration ratio, date size and energy consumption under constant velocity are close to the condition of constant acceleration. EasiAC's active opportunity error under constant velocity is smaller than the condition of constant acceleration. This is because the target motion model is CV model. When the target does constant acceleration linear movement, the movement model of the target is inconsistent with target moving state. In addition, EasiAC's sampling frequency, sampling time, image integration ratio, data size and energy consumption under constant velocity are close



**Fig. 7. The comparison results of four different scenarios.**

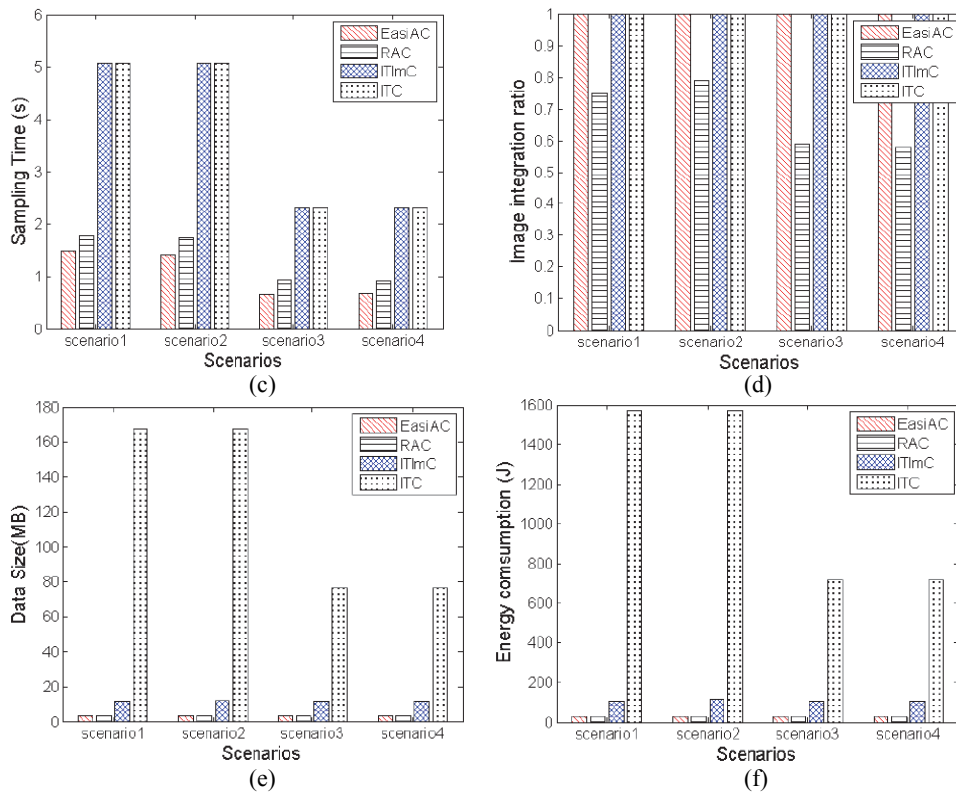


Fig. 7. (Cont'd) The comparison results of four different scenarios.

to the condition of constant acceleration. The experiment results show that the performance of EasiAC is better than the performance of RAC, ITImC and ITC under both constant velocity and constant acceleration.

## 5.2 Real Road Experiments

In this section, we will deploy the magnetic and the camera sensor nodes in real road environment to verify the validity of EasiAC. Specially, we first evaluate the active opportunity, sampling frequency and sampling time when the target is with different velocity. Then the total data size and energy usage are analyzed. What is more, BbTC algorithm's classification accuracy is compared with the traditional methods.

### 5.2.1 Experiment Methodology and Settings

The experiment scenario in our real road experiments is shown in Fig. 8. The operating system used in the sensor node is Tinyos-2.x. Our experiment includes two magnetic sensor nodes, one camera sensor node (Imote2 and IMB400), one clock synchronization node, one sink node, one base station and one portable computer. We use two magnetic sensor nodes to predict the velocity of the targets (*e.g.*, bicycle and car) based



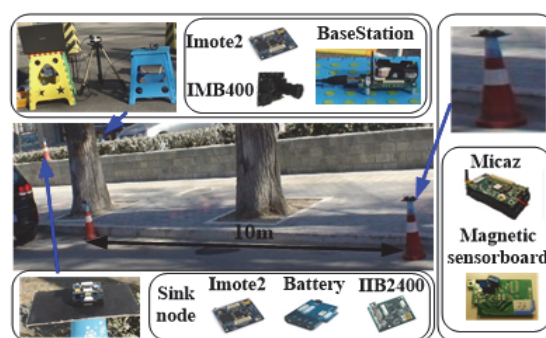


Fig. 8. Real experiment scenario.

on Bayesian filtering. Camera sensor node is used to capture images and send them to the portable computer. Besides, camera sensor node also can make classification online. Clock synchronization node is used for synchronization between all sensor nodes. Sink node is used to receive the features from the magnetic sensor nodes and send the command messages to the camera sensor node. Base station is used to receive the messages and send them to the portable computer and portable computer is used to save all the experiment results. Other parameters settings are the same as the Table 1. Re-synchronous operation needs to be performed every 10 minutes, as the clock drift problem. Thus, ten minutes statistical results are taken as one group.

What is more, we first manually collected sample images as training data for vehicle classification in this experiment. Each time when a vehicle passed the camera's coverage area, we would manually activate the camera to work 1 s, and the image data would be saved to a laptop computer. Throughout this experiment, we collected sample data, including 259 bicycles, 140 cars. The feature extraction and feature selection are discussed in section 4. Besides, we compare the classification accuracy of our approach BbTC with the classification accuracy of the method (DstM method) [15], which weight the local classifier based on the distance between the sensor and the target. In this paper, only one camera sensor node is used to make classification, thus the weight of the local classifiers is the same as each other in [15].

### 5.2.2 Experiment results

We conduct two group experiments online: the first group includes 18 bicycles and 12 cars, and 14 bicycles and 15 cars are in the second group. So there are total of 59 target samples and all the samples are sorted in ascending order according to their velocity. Fig. 9 and 10 show the statistical results of the relationship between the active opportunity, sampling frequency, sampling time and velocity. The results of image integration ratio and classification accuracy are shown Tables 3 and 4 respectively.

As shown in Figs. 9 and 10, EasiAC can adjust the camera sensor's active opportunity, sampling frequency and sampling time adaptively according to the targets' velocity. Sampling frequency varies directly with the velocity, sampling time and active opportunity varies inversely with the velocity. Table 3 shows there are 59 samples, include-

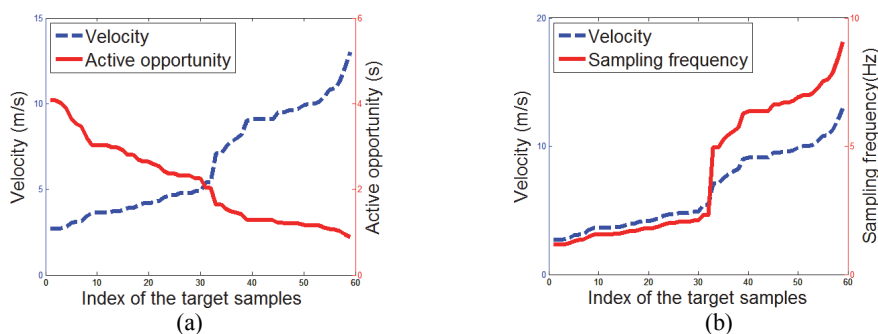


Fig. 9. Active opportunity and sampling frequency vs velocity.

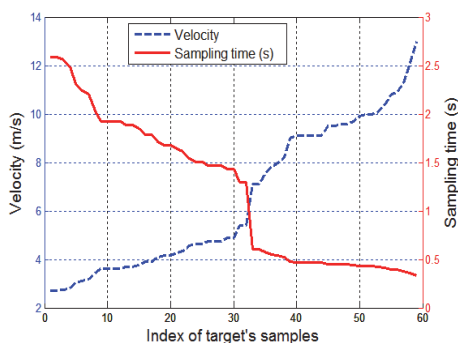


Fig. 10. Sampling time vs velocity.

**Table 3. Results of image integration ratio.**

Category	Results		
	Number of samples	Real type ( $i/f^a$ )	Image integration ratio
bicycle	32	29/3	90.6%
car	27	24/3	88.9%

<sup>a</sup>  $i$  and  $f$  are the number of the complete images and incomplete images respectively.

**Table 4. Results of classification accuracy.**

Method	Type	Results			
		Total samples	Integrated samples	Real type ( $i/f^b$ )	Classification Accuracy
BbTC	bicycle	32	29	29/0	100%
	car	27	24	24/1	95.8%
DstM	bicycle	32	29	29/1	96.6%
	car	27	24	24/2	91.7%

<sup>b</sup>  $i$  and  $f$  are the number of the correctly classified samples and incorrectly classified samples respectively.

ing 32 bicycles and 27 cars. Image integration ratio of bicycle and car are 90.6% and 88.9% respectively. Besides, we use the features LHR and PAR to make classification. The feature abstraction and classification are online. As shown in Table 4, the BbTC's classification accuracy of the bicycle and the car are 100% and 95.8% respectively, while DstM's classification accuracy of the bicycle and the car are 96.6% and 91.7% respectively. Therefore, the result shows that BbTC has higher classification accuracy than

traditional methods such as DstM method. The total data size and energy consumption of the 59 experiments are 64.9MB and 1591.4J respectively in EasiAC, while the traditional method (ITC) are 2336.4MB and 58294J respectively. Therefore, the traditional method's data size and energy consumption are 36 times larger than EasiAC. The results demonstrate that EasiAC generate less data size and consumes less energy consumption than the traditional solutions while maintaining the performance at acceptable level in the presence of dynamics of the monitored target.

## 6. CONCLUSION AND DISCUSSION

The collaboration's performance is inevitably weakened by the dynamics of the monitored target. In this paper, we proposed an adaptive collaboration method EasiAC to solve this problem. It exploits low-quality sensors to predict the target state via Bayesian filtering. Then it calibrates the active opportunity, sampling frequency and sampling time adaptively according to the current state of the target. Finally, we propose BbTC algorithm to make target classification to achieve high accuracy. The results of simulations and real road environment experiments demonstrate that our adaptive collaboration method can effectively adjust its active opportunity, sampling frequency and sampling time, which obtains balance between sensing accuracy and energy-efficiency in presence of the dynamics of the monitored target.

Besides, in section 5.2, we have deployed the magnetic and the camera sensor nodes in real road environment to verify the validity of EasiAC. In real WSN/telematics, there are large amount of magnetic sensor nodes which are used to predict the velocity of monitored targets (e.g., bicycle, car) based on Bayesian filtering. The neighbor magnetic sensor nodes are clustered. Each cluster has one head which is used to receive the sensing information from the magnetic sensor nodes in the cluster. At the same time, cluster head computes some critical parameters (e.g., active opportunity and sampling frequency) and send these messages to the camera sensor node. Camera sensor node is triggered and captures images, and makes classification online when the active opportunity comes. Clock synchronization node is also needed for synchronization between all sensor nodes.

## REFERENCES

1. M. Yarvis, N. Kushalnagar, H. Singh, A. Rangarajan, Y. Liu, and S. Singh, "Exploiting heterogeneity in sensor networks," in *Proceedings of the 24th Annual Joint Conference of IEEE Computer and Communications Societies*, 2005, pp. 878-890.
2. V. Jelici, "Power management in wireless sensor networks with high-consuming sensors," Qualifying Doctoral Examination, 2011.
3. Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *Journal of Computer and System Sciences*, Vol. 55, 1997, pp. 119-139.
4. K. J. Xiao, R. Wang, and L. Cui, "Adaptive collaboration for heterogeneous sensor networks in dynamic environments," in *Proceedings of IEEE Global Communications Conference*, 2013, pp. 146-151.

5. P. Boonma and J. Suzuki, "BiSNET: A biologically-inspired middleware architecture for self-managing wireless sensor networks," *Computer Networks*, Vol. 51, 2007, pp. 4599-4616.
6. A. Munir and A. Gordon-Ross, "An MDP-based dynamic optimization methodology for wireless sensor networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 23, 2012, pp. 616-625.
7. C. Alippi, G. Anastasi, C. Galperti, F. Mancini, and M. Roveri, "Adaptive sampling for energy conservation in wireless sensor networks for snow monitoring applications," in *Proceedings of IEEE International Conference on Mobile Adhoc and Sensor Systems*, 2007, pp. 1-6.
8. Z. Chen, G. Barrenetxea, and M. Vetterli, "Share risk and energy: Sampling and communication strategies for multi-camera wireless monitoring networks," in *Proceedings of IEEE INFOCOM*, 2012, pp. 1862-1870.
9. S. Cheng, J. Li, and R. Yu, "Bernoulli sampling based ( $\epsilon$ ,  $\delta$ )-approximate aggregation in large-scale sensor networks," in *Proceedings of the 29th Conference on Information Communications*, 2010, pp. 1181-1189.
10. R. Tan, Q. Ren, and L. Yu, "Adaptive calibration for fusion-based wireless sensor networks," in *Proceedings of IEEE INFOCOM*, 2010, pp. 1-9.
11. J. Chen, R. Tian, G. Xing, X. Wang, and X. Fu, "Fidelity-aware utilization control for cyber-physical surveillance systems," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 23, 2012, pp. 1739-1751.
12. T. Kijewski-Correa, M. Haenggi, and P. Antsaklis, "Multi-scale wireless sensor networks for structural health monitoring," in *Proceedings of ASCE Structures Congress*, 2006, pp. 16-18.
13. X. Wang, S. Wang, D. Bi, and J. J. Ma, "Distributed peer-to-peer target tracking in sensor networks," *Sensors*, Vol. 7, 2007, pp. 1001-1027.
14. X. Wang and S. Wang, "Collaborative signal processing for target tracking in distributed wireless sensor networks," *Journal of Parallel and Distributed Computing*, Vol. 67, 2007, pp. 501-515.
15. M. Duarte and Y. Hu, "Vehicle classification in distributed sensor networks," *Journal of Parallel and Distributed Computing*, Vol. 64, 2004, pp. 826-838.
16. L. Liu, A. Ming, H. Ma, and X. Zhang, "A binary-classification-tree based framework for distributed target classification in multimedia sensor networks," in *Proceedings of IEEE INFOCOM*, 2012, pp. 594-602.
17. X. Wang, S. Wang, and D. Bi, "Distributed visual-target-surveillance system in wireless sensor networks," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 39, 2009, pp. 1134-1146.
18. A. D'Costa, V. Rama Chandran, and A. Sayeed, "Distributed classification of Gaussian space-time sources in wireless sensor networks," *IEEE Journal on Selected Areas in Communications, Special Issue on Sensor Networks*, Vol. 22, 2004, pp. 1026-1036.
19. T. Takiguchi, R. Takashima, Y. Ariki, and T. Imada, "An AdaBoost-based weighting method for localizing human brain magnetic activity," in *Proceedings of Signal and Information Processing Association Annual Summit and Conference*, 2012, pp. 1-4.
20. P. Panuccio and H. Ghasemzadeh, "Power-aware action recognition with optimal sensor selection: An AdaBoost driven distributed template matching approach," in *Proceedings of ACM Workshop Sensys*, 2011, pp. 1-6.

21. L. Walchshausl and R. Lidnl, "Multi-sensor classification using a boosted cascade detector," in *Proceedings of IEEE Intelligent Vehicles Symposium*, 2007, pp. 1045-1049.
22. J. Shi, X. Sha, Q. Zhang, and N. Zhang, "Extrapolation of band limited signals in linear canonical transform domain," *IEEE Transactions on Signal Processing*, Vol. 60, 2012, pp. 1502-1508.
23. F. Sottile, H. Wymeersch, M. A. Caceres, and M. A. Spirito, "Hybrid GNSS-terrestrial cooperative positioning based on particle filter," in *Proceedings of IEEE Global Communications Conference*, 2011, pp. 1-5.
24. J. Liu, J. Reich, and F. Zhao, "Collaborative in-network processing for target tracking," *Journal on Applied Signal Processing*, Vol. 23, 2003, pp. 378-391.
25. C. E. Shannon, "Communication in the presence of noise," in *Proceedings of IRE*, Vol. 37, 1949, pp. 10-21.
26. M. Buettner, G. Yee, E. Anderson, and R. Han, "X-MAC: A short preamble MAC protocol for duty-cycled wireless sensor networks," in *Proceedings of the 4th International Conference on Embedded Networked Sensor Systems*, 2006, pp. 307-320.
27. R. Wang, L. Zhang, R. Sun, J. Gong, and L. Cui, "EasiTia: A pervasive traffic information acquisition system based on wireless sensor networks," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 12, 2011, pp. 615-621.
28. D. Jung, T. Teixeira, and A. Savvide, "Sensor node lifetime analysis: Models and tools," *ACM Transactions on Sensor Networks*, Vol. 5, 2009, pp. 457-469.
29. R. Wang, L. Zhang, K. J. Xiao, R. L. Sun, and L. Cui, "EasiSee: Real-time vehicle classification and counting via low-cost collaborative sensing," *IEEE Transactions on Intelligent Transportation Systems*, Vol. 15, 2014, pp. 414-424.



**Kejiang Xiao (肖克江)** received the Ph.D. degree in 2015 from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China. He is currently working for the State Grid Information and Communication Company of Hunan Electric Power Company, Changsha, China. His research interests include wireless sensor networks, and information fusion and collaboration.



**Rui Wang (王睿)** received the Ph.D. degree in pattern recognition and intelligent systems from Northwestern Polytechnical University, Xi'an, China, in 2007. He is currently an Associate Professor with the School of Computer and Communication Engineering, University of Sciences and Technology Beijing, Beijing, China. His research interests include wireless sensor networks, intelligent transportation systems, and information fusion.



**Cai Cai (蔡彩)** received her B.A. in English in 2008 and M.A. in English Literature in 2011, both from Hunan Normal University. She is currently working for Changsha Environmental Protection College, Changsha, China. Her research interests include applied linguistics, linguistic data analysis, and translation studies.



**Shaohua Zeng (曾少华)** received the B.S. degree and the M.S. degree from the Wuhan University between 2003 and 2009. He is currently working toward the State Grid Information and Communication Company of Hunan Electric Power Company, Changsha, China. His research interests include VPDN, network switching, routing, and network security.



**Zahid Mahmood** received BS. degree in Computer Science from University of Baluchistan Quetta and MS-Computer Sciences from international Islamic University, Islamabad, Pakistan. His major research area is key management techniques in wireless sensor network and lightweight cryptography techniques for Internet of things (IoT) and authentication, privacy and secure communication for wearable devices. He has been working in International Islamic University, Islamabad after that he join the Mohi-ud-Din Islamic University Nerian Sharif, as a Lecturer. He is currently pursuing the Ph.D. degree in Computer Science at School of Communication Engineering from University of Science and Technology Beijing, China.