Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2014, Article ID 275609, 8 pages
http://dx.doi.org/10.1155/2014/275609

Research Article

Adaptive Collaborative Detection for Opportunistic Vehicle Sensor Networks

Yuanyuan Zeng[1,2]

[1] *School of Electronic Information, Wuhan University, Wuhan 430072, China*
[2] *Su Zhou Institute of Wuhan University, Suzhou 215123, China*

Correspondence should be addressed to Yuanyuan Zeng; zengyy@whu.edu.cn

Received 31 October 2013; Revised 12 April 2014; Accepted 15 April 2014; Published 5 May 2014

Academic Editor: Mohamed Tawhid

Copyright © 2014 Yuanyuan Zeng. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Transportation is a huge problem that curbs development of societies and economy in many countries nowadays. Participatory sensing technologies encourage people to be involved in environment monitoring through smart devices. Vehicle sensor networks (VSNs) are a novel solution for road event detection with advantages. In this paper, we consider runtime road event detection using VSNs to satisfy the application-specific requirements through collaboration among vehicles. A group road detection scheme (GRD) by using dynamic clustering in VSNs is presented to improve detection performance with low time complexity and message complexity. The simulations with testbed of 5 remote controllable vehicles show that GRD scheme provides effectiveness under different road scenarios.

1. Introduction

Transportation problems become more and more imminent in many countries nowadays. Road detection is necessary and important for road safety and good transportation. Participatory sensing encourages people to take part in environment monitoring by smart devices that they carry. It helps to realize pervasive sensing and computing through public monitoring and information sharing.

A vehicle sensor network is a kind of special mobile wireless sensor networks for vehicle environment [1]. Sensor devices equipped on vehicles constitute a vehicle based mobile sensor network. The VSN senses data opportunistically and collects data through ad hoc communications among vehicles. Consider the mobility and coverage of vehicles among a city without extra overhead; a VSN provides a novel and convenient solution for road event detection. Considering high mobility of vehicles, however, collected sensing data in a VSN is often too sparse to use. To improve the detection performance, group detection is a feasible solution based on collaboration among vehicles.

In this paper, we consider runtime road detection in VSNs to improve application-specific detection performance. We present a group road detection scheme (GRD) by utilizing dynamic collaborative clustering. In GRD, the nodes dynamically join and leave the cluster according to the detection requirements. Each node in a cluster collaborates together to detect a road event. The detection is required to achieve application-specific detection performance without much overhead. There is a tradeoff between detection capability and complexity.

The main contribution of this paper includes the following. (1) The novel group detection scheme in VSNs is proposed to improve detection performance. (2) Dynamical clustering algorithms are proposed to form groups and achieve group detection. (3) The proposed algorithms are distributed with low message and time complexity.

Section 2 presents the related work. Section 3 presents the network model. We present the details of GRD in Section 4. Section 5 is simulations and evaluations. Section 6 concludes this paper.

2. Related Works

Participatory sensing is a revolutionary research direction, which allows people with smart device to participate in environment detection and monitoring voluntarily. In this situation, the sensed data will be processed and shared

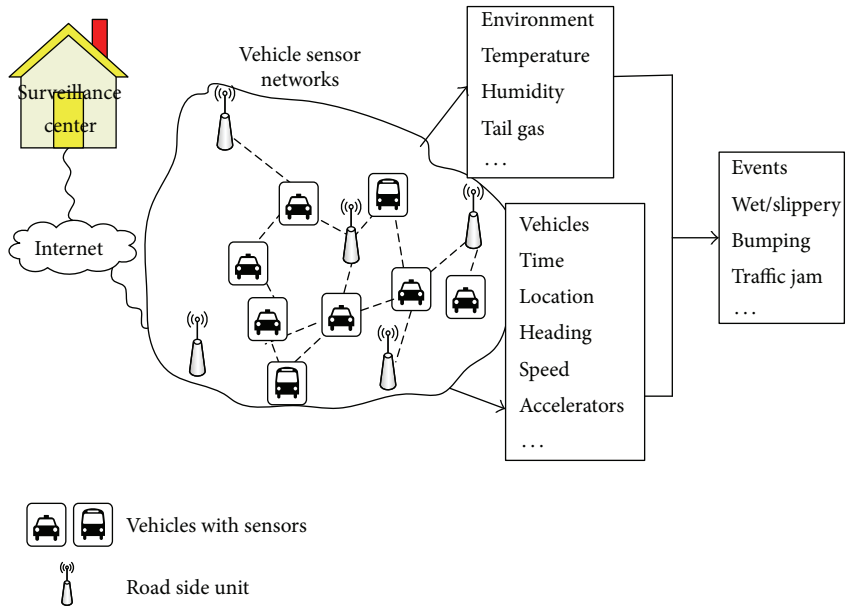


Figure 1: A vehicle sensor network for road event detection applications.

through wireless networks and the other basic infrastructures. Vehicle sensor networks [2] are the combined outcome of vehicle ad hoc networks and WSNs. VSNs could be used in a wide range of applications [3], for example, in traffic monitoring to save communication costs and avoid network congestions. Vehicle based sensors could also be used to monitor and identify possible vehicle thefts by detecting unauthorized vehicle movements.

In recent research work of sensor networks, road surface monitoring related applications are popular. Mohan et al. [4] propose system solution to monitoring road and traffic conditions in a city, which aims to detect potholes, bumps, braking, and honking by using smart phones with the accelerometer, microphone, GSM radio, and GPS sensors. Karuppuswamy et al. [5] propose a vision based scheme for pothole avoidance in a mobile robot by the use of cameras. They detect potholes by looking for large circular objects in the field of view. Eriksson et al. [6] propose road surface pothole detection by using machine learning mechanism based on hidden Markov model. Keally et al. [7] propose a confident event detection algorithm based on hidden Markov model in terms of application-specific detection demand.

For vehicular networks, clustering is a necessary method to cluster and manage vehicles [8]. Rawashdeh and Mahmud [9] propose a speed-overlapped clustering technique on highways by using vehicular speed and relative movement direction. Gurung et al. [10] propose a moving-zone based architecture with moving object modeling and indexing techniques to facilitate message routing in VANETs according to the unique characteristics of network environments. Ucar et al. [11] propose a clustering mechanism to choose the node with the least mobility calculated by speed difference among neighbor nodes as the cluster head to maintain the stable clustering in VANETs. Most of the above work is for vehicular ad hoc networks and the clustering scheme aims to communications among vehicles. None of them is for effective road event detection. Communications focus on the effective connections and information exchange, while detection focuses on the retrieve of useful data among vehicles.

In this context, the paper focuses on dynamically-clustered collaboration based group road detection scheme in vehicle sensor networks with low overhead.

3. System Models

Vehicle sensor networks are widely used in the area of intelligent transportation systems related applications, and here the paper focuses on road event detection applications. Vehicles collect road event data by opportunistic communications as shown in Figure 1.

We use a dynamic graph $G(V, E; T)$ to model the VSNs. Each vehicle $v \in V$ is a graph node. Here each vehicle (graph node) can include more than one sensor to detect the road events. The graph edge $e \in E$ represents the opportunistic wireless communication link between vehicles. Because of the time-dependent opportunistic communication, the connecting time t is used to record the vehicle communication trajectory.

In road detection process, each vehicle can be in one of the states: *initialized*, *clustering*, and *clustered* as in Figure 2.

"*Initialized*" is the initial state for vehicle when it is not in any of the groups. An invoker broadcasts out a cluster-request message to begin the dynamic clustering process.

"*Clustering*" is the state when a node is in the decision of whether to be in the current cluster or not. Either an invoker or the other vehicles can be in this state. If an invoker receives responses from the other nodes that are willing to cooperate, the clustering process is successful and then the invoker enters into "*clustered*." Otherwise, the vehicle will

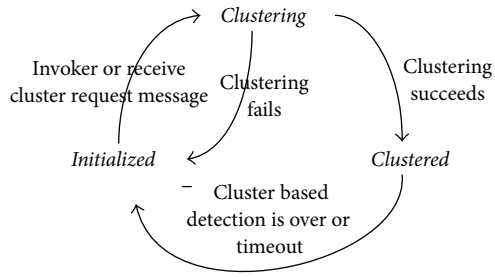


Figure 2: State transition machine for a vehicle.

enter back to "*initialized.*" If it is a noninvoker vehicle that is in "*clustering,*" it will calculate to make decision on whether to join or to reject the clustering request. A noninvoker node which decides to join the current cluster will enter into "*clustered.*" Otherwise, it enters into "*initialized.*"

Obviously, "*clustered*" is the state that a node is in the cluster and cooperates with the other cluster members. When the cluster based group detection is over or larger than a predetermined timeout, the node in "*clustered*" enters back to "*initialized*" state and leaves the cooperation. The *timeout* is used to be released from long time utilization or lost connection from the clustering. If a vehicle joins the group detection, it will keep staying in the group and work as a group member for a while. The maximal working time is *timeout*.

4. GRD Design

GRD design aims to construct dynamic groups (clusters) to guarantee efficient road detection. The dynamic cluster is constructed through the following function modules: local invoker, cluster generation, and runtime decision modules.

"Local invoker" is the module to determine the local invoker for GRD. The local invoker is the vehicle with sparse data that needs to utilize GRD to improve the detection. The invoker sends out a group detection request (i.e., cluster-request message) to request for clustering.

"Cluster generation" is the module to choose event-related vehicles to join into groups and try to satisfy the detection requirement.

"Runtime decision" module makes decisions for the vehicle on whether to join/form the group or not.

Figure 3 illustrates the above modules on both invoker and noninvoker vehicles, respectively. The noninvoker vehicles join the cluster or reject the request through the modules. It can also terminate current cluster based group detection according to application requirements or local detection efficiency.

4.1. The Local Invoker. For the convenience of illustration, some of the notations used in the paper are explained in Notation Section.

Let the road detection zone Z, which can be modeled by an approximate rectangle area and divided into subzone as grids $g_1, g_2, \ldots, g_m$ in the whole surveillance area, that is, $Z = \{g_1, g_2, \ldots, g_m\}$. For detection grid g_i, the centroid location is denoted by z_i. Let $d_{k,i}$ be the distance between a vehicle n_k and event related grid g_i. In (1), $n_k(x, y)$ is the coordinates of vehicle n_k and $z_i(x, y)$ is the coordinates of grid centroid

$$d_{k,i} \equiv d(n_k, g_i) = \left\| n_k(x, y) - z_i(x, y) \right\|. \quad (1)$$

$c(n_k, g_i)$ denotes the static detection capability of vehicle n_k on a specific static position toward the road grid g_i. The vehicle can be equipped with sensors to detect road events. The maximal sensing range of the vehicle n_k is denoted as R_k. A probabilistic model is utilized to calculate the detection capability of the vehicle toward a specific event grid by considering the distance between the vehicle and the grid centroid. It is obvious that a vehicle has better detection capability when it is near the grid. $c(n_k, g_i)$ can be illustrated as follows:

$$c(n_k, g_i) = \begin{cases} \dfrac{1}{1 + \sqrt{d_{k,i}}} & d_{k,i} \le R_k \\ 0 & d_{k,i} > R_k. \end{cases} \quad (2)$$

The vehicle detection capability can be calculated according to (2) as a value between 0 and 1. So, it is convenient for upper-layer applications to give a predefined capability parameter or threshold according to application requirements and long-time observation data.

Considering the vehicle mobility, the mobile detection capability (to abbreviate it as detection capability in the remaining) for vehicles should include the summation of all the static detection capability over time and space. $c'(n_k, g_i)$ is used to denote the detection capability of vehicle n_k toward the road event grid g_i, when it moves in grid g_i. In (3), $(x_{\min}, y_{\min})$ and $(x_{\max}, y_{\max})$ are the coordinate range of the vehicle movement and (t_1, t_2) is the time durations of movement

$$c'(n_k, g_i) = \int_{t_1}^{t_2} \int_{x_{\min}}^{x_{\max}} \int_{y_{\min}}^{y_{\max}} c(n_k, g_i)\, dt\, dx\, dy. \quad (3)$$

If a vehicle moves across the grids, the detection capability for a specific road event is calculated only through the event-related grids during the movement. Figure 4 shows an example. In the example, the vehicle moves across the upper-right side two grids g_4 and g_5. But if only g_5 is event-related grid, the detection capability of the vehicle will be calculated according to the movement within the grid g_5. The event-related grid is decided by the event detection precision that is usually defined by the upper-layer application with a detection precision range (which can be modeled as a circle with a predefined radius). The grids that fall into the precision range are the event-related grids.

If a vehicle finds the local detection capability not suitable for the application-specific detection performance, it will invoke the group detection process and becomes an invoker.

According to the state transition machine, the local invoker broadcasts out a cluster request with the request of clustering process and enters into "*clustering.*" The cluster-request message broadcast is based on CSMA/CA mechanism to avoid contentions. The node in "*initialized*" state

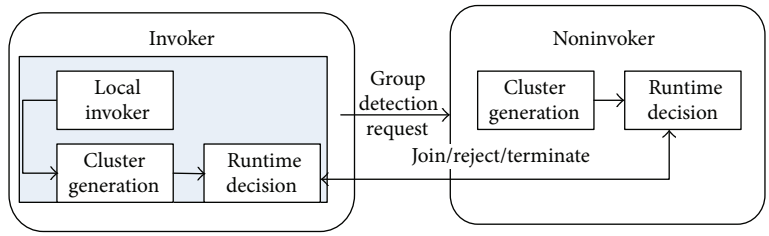


Figure 3: Dynamic cluster related function modules.

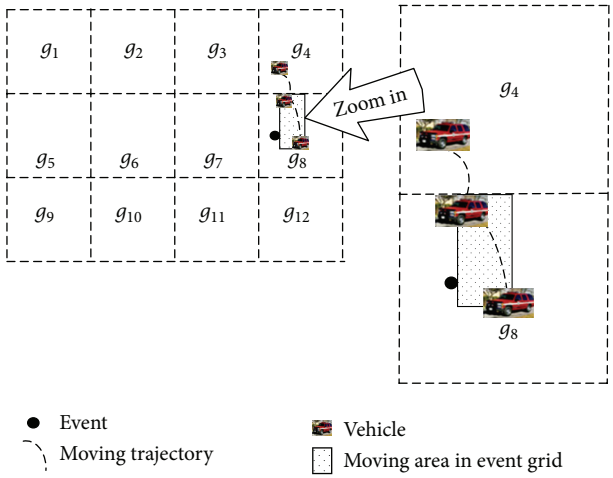


Figure 4: An example of detection scenario when vehicles move across the grids.

that receives a cluster request message enters into "*clustering*" state, which is the state to make decision on cooperation.

Algorithm 1 describes Pseudocode of the invoker selection. The message is broadcasted among the neighborhood. Each node executes the algorithm in sequence in a distributive way. It is obvious that Algorithm 1 is with $O(|\Delta|)$ message complexity and $O(n)$ time complexity, where Δ is the maximal degree of the network graph.

4.2. Cluster Generation. After the invoker broadcasts out a cluster request, the neighbors in "*initialized*" state which receives the message will decide whether to join the requested group detection or not by runtime decision module. If nodes in other states receive the message, they just ignore it. The cluster request message will be forwarded in the network until the group achieves satisfied detection capability.

Define the group detection capability of vehicle group set S invoked by n_k for a specific road event which occurs in grid g_i as $c'(S_k, g_i)$. In (4), $1 - c'(n_j, g_i)$ is the detection disability degree of vehicle n_j toward grid g_i. Then the disability of the set S is achieved by the multiplier of each vehicle's disability. Based on it, the group detection capability is shown as in

$$c'\left(S_k, g_i\right) = 1 - \prod_{n_j \in S_k} \left(1 - c'\left(n_j, g_i\right)\right). \quad (4)$$

Algorithm 2 describes cluster generation process. To curb the number and scale of forwarding messages, a maximal forward number MAX_FW is defined to avoid excessive broadcast messages in the network. It is easy to see that Algorithm 2 is the same as Algorithm 1 with low time/message complexity and suitable for VSNs.

4.3. Dynamically Clustered Cooperation and Decision. In runtime decision process, for an invoker, vehicles that can contribute to current event detection are invited to join in to improve detection capabilities. For a noninvoker, it will make decision according to the detection relevance with the inviter. So it needs a negotiation process between runtime dynamic-clustered cooperation between the invoker and the noninvoker. Through the negations among vehicles, nonrelevant vehicles are filtered from the cluster and relevant vehicles are involved in the cluster to make group detection with the application-specific detection capability.

The detection relevance metric is defined to determine what kind of degree the vehicle can contribute to the current event detection. Assume invoker n_k and "*clustering*" vehicle n_j get observed dataduring time duration T from start time t_1 to end time t_2. The detection relevance of n_j with cluster C_k toward event grid g_i is denoted as $r_i(C_k, n_j)$, which is defined as shown in (5), in which, $c'(C_k, g_i)$ is the detection capability of cluster C_k toward event grid g_i during time duration T, and $c'(n_j, g_i)$ is the detection capability of vehicle n_j toward g_i during T

$$r_i\left(C_k, n_j\right) = \frac{c'\left(C_k \cup \left\{n_j\right\}, g_i\right)}{c'\left(C_k, g_i\right)} \quad \text{when } c'\left(C_k, g_i\right) > 0. \quad (5)$$

Especially, before the cluster is constructed, the detection relevance of n_j with n_k toward event grid g_i is denoted as $r_i(n_k, n_j)$ as in

$$r_i\left(n_k, n_j\right) = \frac{c'\left(\left\{n_k\right\} \cup \left\{n_j\right\}, g_i\right)}{c'\left(n_k, g_i\right)} \quad \text{when } c'\left(n_k, g_i\right) > 0. \quad (6)$$

If $r_i(C_k, n_j) > 1$ or $r_i(n_k, n_j) > 1$, it implies that n_j will contribute to the event detection for current cluster or possible cluster. In this way, the detection relevance can be calculated. The relevant vehicles join into the cluster or leave the cluster to achieve efficient detection. Algorithm 3 is the pseudocode of runtime decision. It is obvious to see that the algorithm is with linear message complexity and time complexity.

```
Input: a VSN G(V, E; T), application-specific detection capability c_th, initial state "initialized"
Output: nodes for clustering process
for each node in G
   If (c'(g_i, n_k) > 0)(c'(g_i, n_k) <  c_th) && (state = "initialized")
       Ignore received messages
       broadcast a Cluster-Req message
       state ← "clustering"
   elseif receives a Cluster-Req && (state = "initialized")
       state ← "clustering"
   endif
endfor
```

Algorithm 1: Invoker algorithm.

```
Input: a VSN G(V, E; T), application-specific detection capability c_th, nodes for clustering by Algorithm 1
Output: Set of clusters C
for each node in G
  If (state = "clustering") && (non-invoker) && (receive "join" msgs)
 // Runtime_Decision() is the Runtime Decision algorithm
       state ← "clustered"
       become a cluster member
  elseif (!Runtime_Decision())
 // reject the clustering or terminate from the cluster
       state ← "initialized"
       If (msg_seq <= MAX_FW)
              forward the Cluster_Req msg
       endif
  endif
  If (state  = "clustering") && (invoker)
       If (wait(T_MAX1)) && (receive "join" responses)
  // wait(T_MAX1) is the delay function for clustering
 // T_MAX1 is the maximal waiting time for clustering
              state ← "clustered"
              become a cluster
              If c'(C_k, g_i) < c_th
// C_k is the group set of current cluster invoked by n_k
                Broadcast a Cluster_Req msg
              endif
       elseif (!wait(T_MAX1)) ‖ (receive "reject")
              state ← "initialized"
 // no "join" received during waiting time implies failure
       endif
endif
   If (state = "clustered") && (invoker) && (receive "join"/"terminate" msgs)
       Update the cluster
   endif
endfor
```

Algorithm 2: Cluster generation.

5. Simulations

The overall goal of this part is to make simulations in the remote-control vehicle based transportation situations and then make evaluations to testify the effectiveness and accuracy of GRD. Actually, the detection performance is related to the detection methods. The simulations are implemented based on the same basic detection method with the situation of only local vehicle detection involved and with GRD, respectively. The comparisons are made that try to show the difference between the two schemes based on the same basic detection method. To simplify the tests and show the obvious differences from the results, a threshold based event detection method is taken in the simulations. In the simulations, local vehicle detection scheme is to make basic threshold based event detection without the group collaboration among

Input: a VSN $G(V, E; T)$, application-specific detection capability c_{th}, Set of Growing clusters C (the Step output of Algorithm 2)
Output: node responses for clusters
for each node in G
  If ($state$ = "*clustering*") && (($r_i(C_k, n_j) > 1$) ‖ ($r_i(n_k, n_j) > 1$))
    Response to invoker with msg(*join*)
  else
    Response to invoker with msg(*reject*)
  endif
  If ($state$ = "*clustered*") && application needs
// cluster members can leave the group for application needs
    Response to invoker with msg(*terminate*)
  endif
endfor

Algorithm 3: Runtime decision algorithm.

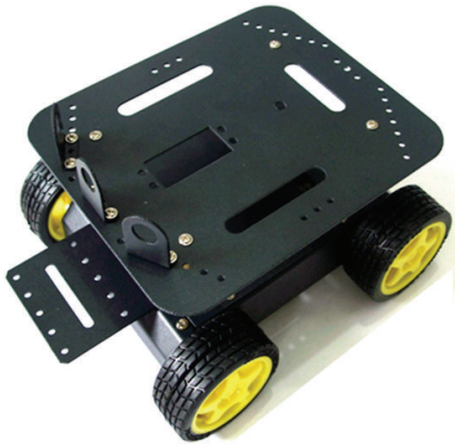


Figure 5: A remote controllable vehicle.

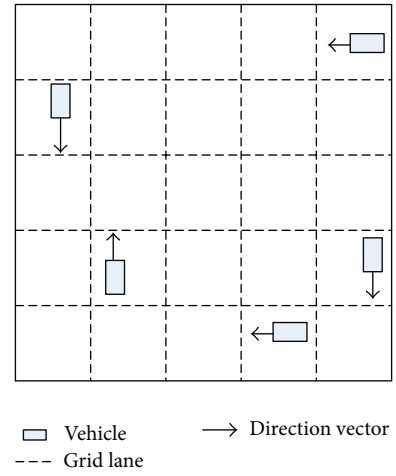


Figure 6: An Illustration of driving scenario in testbed.

vehicles; that is, the sensing data of each vehicle is compared with the application-specific threshold and then to decide the event based on it. The GRD scheme proposed in the paper is to make threshold based event detection with the group collaboration scheme; that is, the average sensing data of the group achieved is compared with the application-specific threshold and then to decide the event.

A simulated testbed consists of 5 remote controllable vehicles. The vehicle can work over 30 minutes continuously. The vehicle for testbed is shown in Figure 5.

An illustration of testbed is shown in Figure 6. The vehicle is based on Arduino control board with 4 wheel drive. It is equipped with accelerometers during the simulations. The vehicle could move flexibly as in real transportation. The sensors on the vehicles are connected to one another wirelessly through the low-power radio. Two vehicles in the testbed are also equipped with a WiFi card. The test area is covered by WiFi. Then the sensing data collected by each vehicle is forwarded and then gathered to central gateway through WiFi connection. The driving area is divided into virtual grids and vehicles drive according to the grid lane. Each vehicle can drive toward the vertical and horizontal directions as shown in the figure. The vehicle can be controlled to change the direction when it approaches the area border or to avoid collisions.

Each vehicle is placed randomly on different locations and lanes and then controlled to drive in different directions around the test area. There are several artificial upward/downward slopes built by boards and slippery road area made by spilling jelly as well as uneven areas with broken stones in the test place to simulate complex road driving. The artificial experiment scenarios are used to simulate uneven, wet, and slippery road conditions. The location of simulated road events is randomly chosen. Vehicles are controlled to move around in the test place. The sensors are mounted on each vehicle with the same orientation and place. The ultrasonic wave sensor is used for measuring the distance and location. In the simulations, the vehicles are controlled to travel within 5 different road condition scenarios as shown in Table 1.

Figure 7 shows the detection accuracy rate within different scenarios. The results show we achieve 25.3%–40.2% more accuracy ratio when using group detection. In GRD, more relevant vehicles are involved into detection and improve

Table 1: Simulated road scenarios.

Scenarios	Number of uneven areas	Number of slippery areas	Number of slopes
Scenario 1	5	2	1
Scenario 2	2	5	1
Scenario 3	8	1	2
Scenario 4	1	8	2
Scenario 5	5	5	3

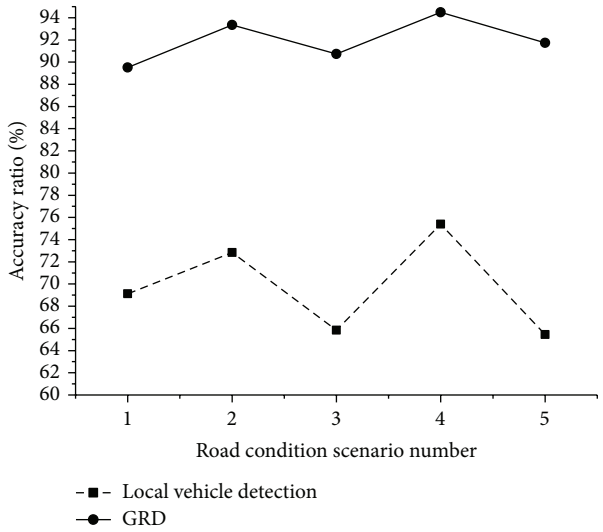


Figure 7: Detection accuracy ratio within different scenarios.

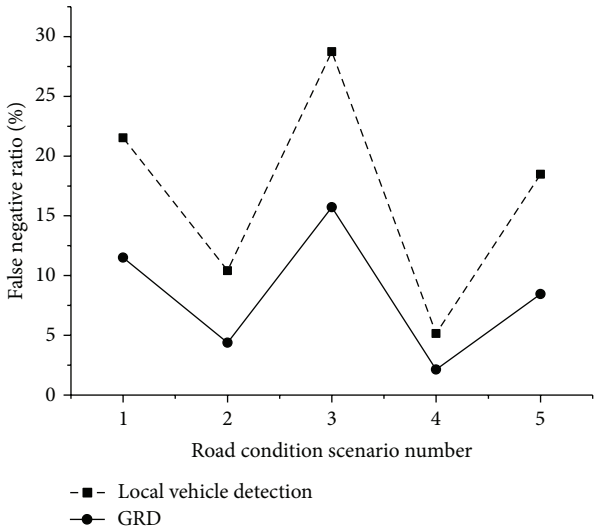


Figure 9: False negative rate within different scenarios.

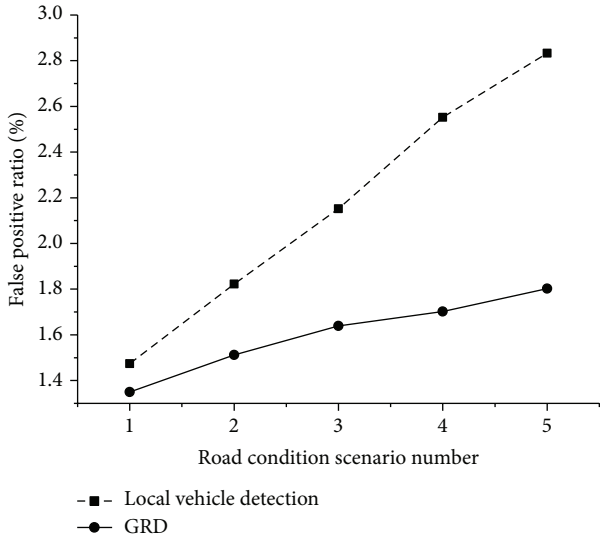


Figure 8: False positive ratio within different scenarios.

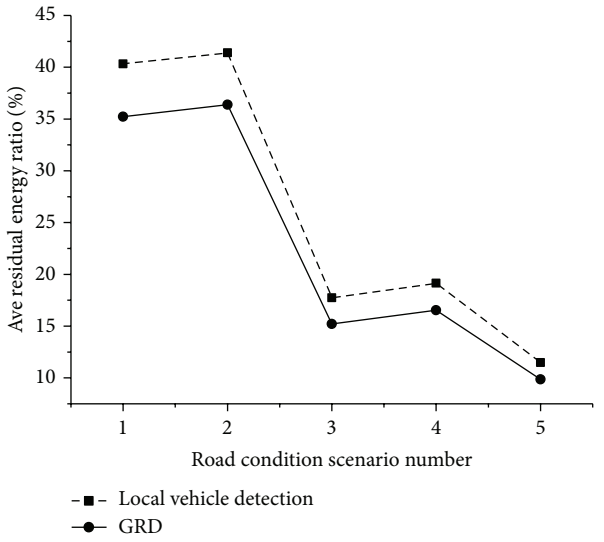


Figure 10: Average residual energy ratio within different scenarios.

the accuracy by more useful observation data. GRD helps to improve the accuracy of detection.

Figure 8 shows false positive rate within different scenarios. GRD helps to improve the false positive rate.

Figure 9 shows the false negative rate within different scenarios. GRD decreases around 52% false negative. Obviously, group detection has better performance than only using local vehicle detection.

Figure 10 shows the average residual energy ratio of network nodes within different scenarios. The residual energy ratio is calculated by residual energy versus initial energy. We compare GRD with local vehicle detection. GRD results in no

more than 17% average residual energy ratio when compared with local vehicle detection cases in the network. So, GRD will not incur too much energy consumption. Especially, for real vehicles, the sustainable energy supply is not a big problem.

6. Conclusion

Vehicle sensor networks will be widely utilized in intelligent transportation systems in a very near future, which can be used to inform the drivers about kinds of road condition events. In this paper, we propose a group detection scheme based on dynamically clustered cooperation scheme called GRD in VSNs. GRD scheme includes algorithms of cluster invoked process, cluster generation, and runtime decision through vehicle collaborations. The simulations are carried out on 5 controllable vehicles with different scenarios. The results show that GRD can help to improve detection performance effectively without much overhead.

The further work will also be carried out on the testbed consisting of voluntary cars equipped with sensors on campus road areas. The test results will also be compared with other related work to testify the effectiveness.

Notation

Z:	The approximated rectangle road detection zone
g_i:	The divided subgrid in Z
z_i:	The centroid location of g_i
n_k:	The vehicle with its ID as k
R_k:	The sensing range of n_k
$d_{k,i}$:	The distance between n_k and g_i
$c(n_k, g_i)$:	The static detection capability of n_k toward g_i
$c'(n_k, g_i)$:	The mobile detection capability of n_k toward g_i
S:	The vehicle group set
C_k:	The cluster invoked by n_k
$c'(S_k, g_i)$:	The detection capability of S invoked by n_k toward g_i
$r_i(C_k, n_j)$:	The detection relevance of n_j with C_k toward g_i.

Conflict of Interests

The author declares that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Thanks are due to the anonymous reviewers and the academic editor. This work was supported by Chinese National Science Foundation Project (no. 61103218), The Natural Science Foundation of Jiangsu Province Youth Project (no. BK2012200), Hubei Province National Science Foundation Project (no. 2011CDB446), and Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry.

References

[1] M. Gerla and L. Kleinrock, "Vehicular networks and the future of the mobile internet," *Computer Networks*, vol. 55, no. 2, pp. 457–469, 2011.

[2] K. Lan, C. Chou, and H. Wang, "Using vehicular sensor networks for mobile surveillance," in *Proceedings of the 76th IEEE Vehicular Technology Conference (VTC Fall)*, pp. 1–5, Quebec City, Canada, September 2012.

[3] H. Hartenstein and K. Laberteaux, *VANET Vehicular Applications and Inter-Networking Technologies*, John Wiley & Sons, 2010.

[4] P. Mohan, V. N. Padmanabhan, and R. Ramjee, "Traffic-sense: rich monitoring of road and traffic conditions using mobile smartphones," Tech. Rep. MSR-TR-2008-59, Microsoft Research, 2008.

[5] J. Karuppuswamy, V. Selvaraj, M. M. Ganesh, and E. L. Hall, "Detection and avoidance of simulated potholes in autonomous vehicle navigation in an unstructured environment," in *Intelligent Robots and Computer Vision XXI: Algorithms, Techniques, and Active Vision*, vol. 4197 of *Proceedings of SPIE*, pp. 70–80, November 2000.

[6] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The Pothole Patrol: using a mobile sensor network for road surface monitoring," in *Proceedings of the 6th International Conference on Mobile Systems, Applications, and Services*, pp. 29–39, June 2008.

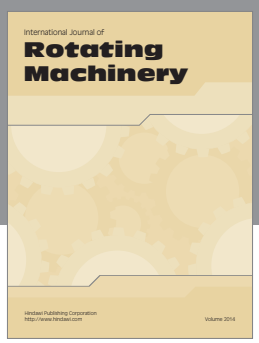
[7] M. Keally, G. Zhou, and G. Xing, "Watchdog: confident event detection in heterogeneous sensor networks," in *Proceedings of the 16th IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS '10)*, pp. 279–288, April 2010.

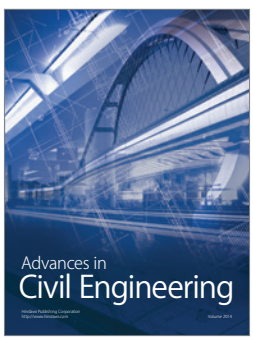
[8] S. Vodopivec, J. Bester, and A. Kos, "A survey on clustering algorithms for vehicular Ad-Hoc networks," in *Proceedings of the 35th International Conference on Telecommunications and Signal Processing (TSP '12)*, pp. 52–56, July 2012.

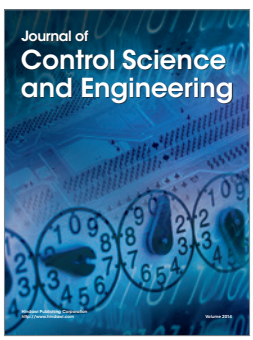
[9] Z. Rawashdeh and S. Mahmud, "A novel algorithm to form stable clusters in vehicular ad hoc networks on highways," *EURASIP Journal on Wireless Communications and Networking*, vol. 2012, article 15, pp. 1–13, 2012.

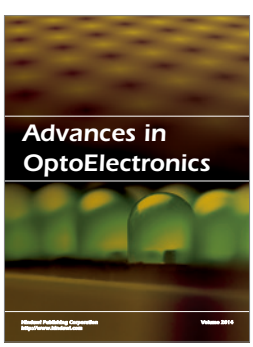
[10] S. Gurung, D. Lin, A. Squicciarini, and O. K. Tonguz, "A moving zone based architecture for message dissemination in VANETs," in *Proceedings of the 8th International Conference on Network and Service Management, Workshop on System Virtualization Management*, pp. 184–188, October 2012.

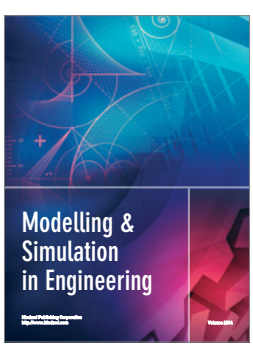
[11] S. Ucar, S. C. Ergen, and O. Ozkasap, "VMaSC: vehicular multi-hop algorithm for stable clustering in vehicular Ad Hoc networks," in *Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC '13)*, pp. 2381–2386, April 2013.

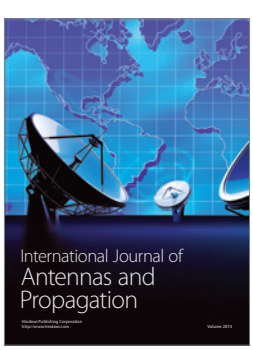


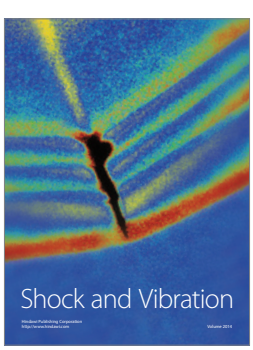


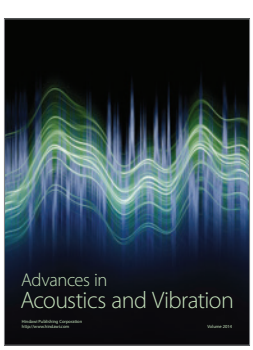












Hindawi

Submit your manuscripts at
http://www.hindawi.com