On Developing Smart Transportation Applications in Fog Computing Paradigm

Nam Ky Giang
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, Canada
kyng@ece.ubc.ca

Victor C.M. Leung
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, Canada
vleung@ece.ubc.ca

Rodger Lea
Department of Electrical and Computer Engineering
University of British Columbia
Vancouver, Canada
rodgerl@ece.ubc.ca

ABSTRACT

Smart Transportation applications by nature are examples of Vehicular Ad-hoc Network (VANETs) applications where mobile vehicles, roadside units and transportation infrastructure interplay with one another to provide value added services. While there are abundant researches that focused on the communication aspect of such Mobile Ad-hoc Networks, there are few research bodies that target the development of VANET applications. Among the popular VANET applications, a dominant direction is to leverage Cloud infrastructure to execute and deliver applications and services. Recent studies showed that Cloud Computing is not sufficient for many VANET applications due to the mobility of vehicles and the latency sensitive requirements they impose. To this end, Fog Computing has been proposed to leverage computation infrastructure that is closer to the network edge to compliment Cloud Computing in providing latency-sensitive applications and services. However, applications development in Fog environment is much more challenging than in the Cloud due to the distributed nature of Fog systems. In this paper, we investigate how Smart Transportation applications are developed following Fog Computing approach, their challenges and possible mitigation from the state of the arts.

Keywords

fog computing; smart transportation; programming model; internet of things

1. INTRODUCTION

Building VANET applications is not an easy task given the large scale of the system and the dynamic nature of its entities such as people, cars or street lanes. Unfortunately, research into Vehicular Ad-hoc Network applications in general is still focused more on communication layer where specific applications are built leveraging some advanced communication technologies. A particular application model or programming abstraction for VANET applications is yet to available. Albeit there are few efforts in applications models and programming abstractions to develop VANET applications, they generally focus on end user experiences such as safety driving, driving assistant or in-car comfort. For example, the authors in [16] proposed to use Mobile Agent model to build a vehicular social network that connects drivers together. Another research direction leveraging Cloud computing as the execution platform for VANET applications with the intuition that scalable Cloud technology will suffice the large scale deployment of many VANETs. However, many VANET applications impose a low latency requirement that are difficult to met using distant Cloud infrastructure. One example application is to use the CCTVs network to count the cars on the street and use the number to control the traffic flow accordingly. In this example, the computation activity that processes the camera stream to count the cars need to reply in a quick manner so that traffic control can be carry out efficiently. With the large scale deployment of CCTVs in transportation infrastructure, relying on a centralized and distant Cloud platform is simply not enough.

Recent advances in communication technologies have facilitated the connectedness in smart transportation components such as vehicles, roadside units and other transportation infrastructure. Along with the Internet of Things direction where there are more and more physical things being connected to the Internet, this connectedness in transportation infrastructure is going to pave the way for new VANETs applications. Thus, investigation into application models and programming abstractions for Smart Transportation applications is an important research direction.

While it is generally agreed that VANET applications can be classified into: comfort applications, road traffic management, traffic coordination and public safety [24], all these classes center around the communication between vehicles to vehicles (V2V) and vehicles to infrastructure (V2I) with end users being the focus in the application logic. We consider in our work a new class of VANET applications called Smart Transportation applications that leverages both vehicles data and data and computing resources from the transportation infrastructure to help in the decision making process. For example, transportation infrastructure such as camera network and roadside units can help autonomous vehicles to navigate their way around. This application does not necessarily involve end users as an integral part of the logic however it does take into account the participation of

DIVANet'16, November 13-17, 2016, Malta, Malta

DOI: http://dx.doi.org/10.1145/2989275.2989286

the vehicles in an autonomous manner. There might be overlapping characteristics when comparing with other classes of VANET applications however, the presence of Fog Computing model with distributed and heterogeneous computing infrastructure and autonomous vehicles raises new challenges that ordinary VANET applications have not met.

In this work, we look at the process of developing Smart Transportation applications from the developers point of view, which is a top down approach in compare to a bottom up approach where advance technologies in communication are exploited in realizing particular applications. In this manner, application models - which represent how Smart Transportation applications are composed and programming abstractions - which support developers in the development process are important and of interested. In general, Smart Transportation applications can be seen as Distributed System applications as the VANET infrastructure consists of distributed devices that are capable of carrying computing resources, e.g road-side units, lamp posts, base station, etc. Existing models and techniques in building Distributed System applications such as message-passing and data-sharing models can be applied, however the challenges in Smart Transportation applications lie in the dynamic nature of vehicles and the heterogeneity of the infrastructure.

Therefore, we argue that developing Smart Transportation applications following a Fog Computing model is not straightforward and is a problem worth exploring. This paper aims at providing an overview of requirements of an application model for Smart Transportation applications. We introduce some existing programming models for this class of applications and open some research directions that are important to achieve the Fog-based Smart Transportation vision.

This paper is organized into 6 sections. In section II we describe the development of Smart Transportation application based on Fog Computing model, we revisit the motivation of Fog Computing and how Fog Computing model is applied in building Smart Transportation applications. Section III discusses about the requirements that a Smart Transportation application platform needs to support in order to make it accessible and usable for general developers. Section IV addresses the challenges that we found crucial in building such an application platform for Smart Transportation systems. Section V reviews the state of the arts in employing Cloud and Fog Computing model as well as available application models and programming abstractions for Fog-based Smart Transportation applications. Finally, section VI concludes our paper.

2. DEVELOPING FOG-BASED SMART TRANSPORTATION APPLICATIONS

2.1 Fog Computing Model

Centralized Computing such as Cloud Computing has been used extensively to support and deliver VANETs and Transportation application in general. However, the exponentially increasing number of connected vehicles poses a great challenge on the Cloud and networking infrastructure in transferring and processing transportation related data such as CCTV streams or road sensory data. Due to the latency-sensitivity and the high volume nature of transportation data, many applications in this area also require a distributed

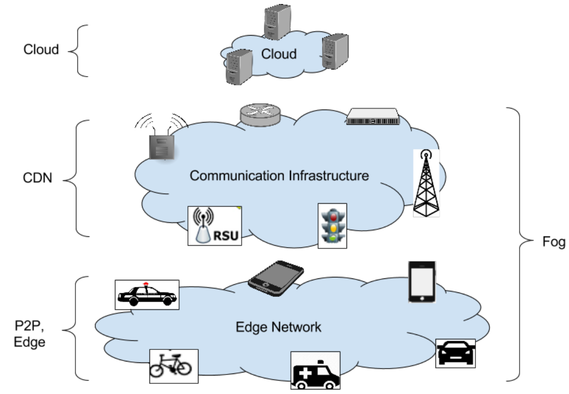

Figure 1: Fog Computing in today's Internet

data processing model rather than a centralized one [13]. For example, driving in urban area usually requires the driver to make decision quickly, such as whether or not to change lane, change route to avoid traffic jam, or to slow down for a parking spot. In order to aid the driver on the decision making process, an application needs to gather many related data such as location, speed of drivers, traffic flow or accident events. It also needs to process those data and react in a real-time manner. This scenario raises a challenge for the Cloud infrastructure in fulfilling the requirements as it has to collect and process a high volume of data in a short amount of time. Having a distributed data processing infrastructure greatly reduces this burden of the Cloud while still achieves the latency-sensitivity requirement.

In an effort to address such shortcomings of the Cloud for the transportation system, researchers have proposed Fog computing [2]. The general idea of Fog Computing is depicted in Fig. 1. Unlike Cloud based resource utilization, the Fog computing paradigm leverages the computing, storage and network resources within and at the edge of the network to augment the capabilities of the Cloud. These processing elements, running on a variety of devices such as network gateways, edge devices etc. provide a mechanism to move processing closer to the network edge. By distributing computing resources closer to users and things, the Fog computing model can be a better choice for building distributed applications [30] [11], especially for latency sensitive applications such as Smart Transportation applications [26] [8].

2.2 Fog-based Smart Transportation Applications

There have been several researches that discuss the suitability of Fog Computing model in VANETs [19]. In the original Fog Computing work [2], the authors identified VANETs applications as one of the main application domains where Fog Computing model shows the best fit. This is particularly true following the recent advancements in communication technologies that allow vehicles and infrastructure to be fully connected to the Internet. In Fog Computing, computing resources at or near the network edge are exploited to deliver latency sensitive applications. This compliments the Cloud Computing in processing tightly coupled sense-actuate loops such as traffic lights control, while the Cloud infrastructure can be used to process data for long-term pur-

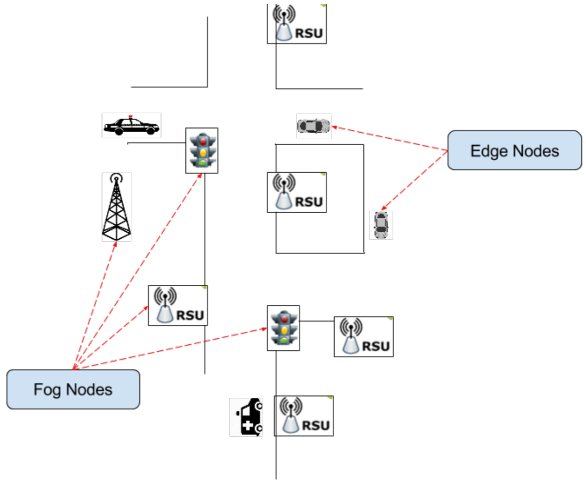

Figure 2: Fog-based Smart Transportation Systems

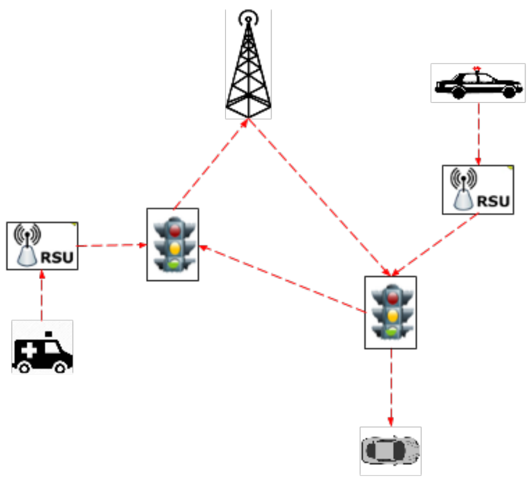

Figure 3: Sample model for Smart Transportation Application

poses such as history analysis or building data models.

While this being promising, there are few researches that explore the development process of a Fog-based Smart Transportation application or a Fog-based VANETs applications from the developers' point of view. Following this top-down approach, we look at the process of software creation where we explore existing application models and programming abstractions that are available to developers. We also look at how realistic it is to transition from such models and abstractions to real world deployment and what platform supports are required.

Based on Fog Computing model, transportation authorities can exploit existing computing infrastructure in the edge or access network to deliver latency sensitive applications. Fig. 2 illustrates how Fog Computing model is applied to Smart Transportation systems where roadside units and transportation infrastructure can involve in the computation process of an application. Meanwhile mobile vehicles can represent the Edge nodes in the network architecture.

There are generally two areas where the development of Smart Transportation applications take place: 1) choosing an application model and programming abstract and 2) which platform supports are required to deploy such applications. In section 3 we discuss these processes in detail, from the requirements for a Smart Transportation application platform to some example applications. Here we give some example applications that can benefit from a Fog-based Smart Transportation system.

2.3 Example Applications

A typical example for Fog-based Smart Transportation applications would be adaptive traffic lights coordination. That is, smart traffic lights should be able to adapt themselves to the real-time traffic circumstances within a particular area. As with the low latency requirement discussed above, the reaction time for one or several smart traffic lights is so short so that it's virtually impossible to offload all the application execution to a distant Cloud. Therefore, such traffic lights should be programmed in a way that they autonomously cooperate with each other and with all the locally available computing resources such as roadside units to coordinate their operations.

Another example of Fog-based Smart Transportation applications is the one found in [13], but with a slightly advancement. In [13], the authors discussed a vehicular search application where CCTVs network in a transportation infrastructure can be used to search for a suspect driving in a vehicle. We suggest that a more advanced version of the same application can also be realized using a crowd-sourced network of dash cams mounted on moving vehicles. In [13], the authors proposed a programming model where an application can be distributed onto a wide range of devices such as CCTV camera, roadside units and the alike. Images from CCTVs are sent to nearby or local computing resources such as a roadside unit to recognize the vehicle in the frame. If the recognition is not successful, the computing unit could issue a Pan-Tilt-Zoom command for the camera to specifically look to a region to get a better image of the suspected car.

In both examples, it is perceived that Smart Transportation applications usually work with stream of data from physical environment. For example, the presence of an emergency vehicle is calculated based on its GPS data or its siren sounds; the plate number is calculated from a stream of image frames from vehicles' dash cams. This is fundamentally different with end users-facing VANET applications where the applications mainly work with service requests from the users. Fig. 3 shows this important application model of Smart Transportation system where the collecting, processing and disseminating of data flows between entities are of paramount important.

These example applications show that there is a need for a software architecture that operates well in Fog-based Smart Transportation systems that is easy and accessible to general application developers. In the next section, we address the requirements of such a software architecture for Smart Transportation applications. The requirements are perceived from our experiences in analyzing Smart Transportation applications as well as from the literature research.

3. SMART TRANSPORTATION APPLICATIONS REQUIREMENTS

3.1 Requirements in application models and programming abstractions

3.1.1 Modular and Reusable

Smart Transportation applications are usually large and complex as it involves heterogeneous devices with many types of data that need a various set of processing algorithms. This characteristic requires an application platform that is modular and flexible so that applications can be incrementally deployed while partial failures do not affect the whole systems. Modular design also allow different data processing algorithms to built and plugged into the system with minimal effort. This is important due to the variety of data streams generated in a Smart Transportation application. Thus the application development process can be done in two independent stages, developing individual modules and developing module interconnection logics. The earlier can be done by component or module providers while the later can be done by Smart Transportation developers.

3.1.2 Scalable

Smart Transportation applications need to be able to work with a large number of vehicles and distributed computing infrastructure so that the supporting platform is required to be scalable. Cloud Computing is a scalable infrastructure however, due to the large amount of real-time data generated by the environment, it might not able to meet the Smart Transportation applications' requirement w.r.t the low latency requirement. An example is the use of CCTV camera network in searching for vehicles using vision processing algorithms as discussed in [13]. In this work, the authors raised the need for a scalable application framework that can handle vision processing of large scale CCTV camera network in a city's transportation infrastructure. While the authors proposed a dynamic scaling solution by merging and spliting Fog processes based on the load, the components involved (the CCTV camera network) is rather fixed. We envision that a more flexible infrastructure where computation resources in dynamic objects such as moving vehicles can also participate in the application.

3.1.3 Context-aware

Since the vehicles and the computing infrastructures are mobile and distributed in large physical areas, Fog-based Smart Transportation applications tend to have logic that involves the vehiclesâĂŹ physical locations [13] or more generally the device's physical context. Thus, it should be necessary that context information be exposed to developers so that they can build context-aware applications. Context-aware or mobility-aware requirements have been introduced in some researches into WSN, especially the case of mobile sensor nodes such as to monitor the activities of wild animals [21]. Some researches also proposed the mobility of sensor sinks where the data sink in a WSN can move [27]. Nonetheless, the exploration in these areas is not common in WSN research and it usually focuses on the communication issue when dealing with mobility requirement, rather than issues on the application layer [23], such as application models and programming abstractions.

3.1.4 High Level Abstraction

Due to the high level of heterogeneity and the large number of devices in a Smart Transportation application, developing Fog-based Smart Transportation applications often requires a very high level of abstraction of how the heterogeneous computations are described, coordinated or interact with one another. That is, in such environment, an application model that demands developers to explicitly specify which device (e.g there needs a deviceâĂŹs IP address in the code) are involved in the logic is not suitable. In another word, an application should not deal with a particular vehicle by its identification. Instead, the model should allow it to work with a pool of many vehicles at once. For example, such a programming abstraction should be able to describe the following need: "get the speed of all the car in this location".

3.2 Requirements in platform supports

3.2.1 Support Low Latency Characteristic

Vehicles on the road usually drive with a speed up to 60 km/h in urban setting and over 100 km/h on highways. If we take an example of an autonomous traffic light that reacts to a coming emergency vehicle. If the vehicle is approaching the traffic light at the distance of 100m and the speed of 60 km/h, assuming yellow light takes 5s to turn into red, the react time required for the traffic light to switch once the vehicle is nearby is roughly 1 second. Although there are many techniques proposed for such an application such as using radio frequency to control the traffic light, we consider in this research an "IoT-style" solution where devices such as vehicles and infrastructure are connected to the Internet and be controlled via the Internet. If Cloud Computing is used to coordinate such application, the round trip time of a network packet alone to reach the Cloud and back already takes around 500ms-1s over wireless settings such as LTE. Therefore, the low latency characteristic of Smart Transportation applications put a requirement on the application framework, especially the communication network and the coordination of communication among components.

3.2.2 Decentralized Computation

In Smart Transportation applications, decentralized execution or programming in the large [6] model is preferable [2]. This is because the applications usually operate over a large number of heterogeneous and dynamic components such as mobile vehicles or roadside units. A centralized application has to implement all sorts of conditions and exception handling to deal with such heterogeneity and dynamic nature. Rather than that, it is more scalable if the application can be developed in a modular way with components being distributed to the devices. In this way, those devices can autonomously cooperate with one another to fulfill the applicationâĂŹs need following a choreography model [5]. In addition, the decentralized approach can leverage computing resources of the transportation infrastructure for the execution of the application instead of relying on a distant Cloud in order to fulfill the latency requirement of the application.

3.2.3 Dataflow-based communication

Smart Transportation applications can be seen as Distributed Systems applications in general where participating devices and computing resources communicate with one another via message passing or data sharing model. However, unlike the conventional Distributed Systems programming, in Smart Transportation, the communication model is more like the exchanging of data flows rather than exchanging of request-response messages. That is, in request-response messaging model, a response is always expected by the sender, either synchronously or asynchronously while in data flow model, the messages tend to flow in unidirectional streams. There could be two unidirectional streams flowing between two entities to represent a two way communication channel, however, there is no waiting of response in either way. In a preliminary work [10], we found that dataflow model is a natural fit for Fog-based applications because of the focus on the need to process a large number of data streams generated from many devices.

4. RESEARCH CHALLENGES

Fog-based Smart Transportation Applications are Distributed System Applications in nature. However, Smart Transportation Systems are much more complicated than a large class of Distributed Systems. Among many difficulties, heterogeneity, dynamic nature and large scale deployment seem to be the unique characteristics of a Smart Transportation system. In this section we discuss these unique characteristics of a Smart Transportation system and raise the challenges that are difficult to solve using existing approaches.

4.1 Heterogeneity in Smart Transportation systems

In researches into WSN and IoT, heterogeneity has been observed as the different among devices with regard their communication protocols or their sensing capability [10]. The authors argued that there exists another dimension of heterogeneity that represents the different in computing capability of devices when looking from a Fog Computing perspective. This is a valid concern since Smart Transportation systems often consist of many different types of devices with different computing capabilities such as roadside units, lamp posts or even vehicles on the road. Thus, Smart Transportation applications need to be aware of the underlying heterogeneous computing platforms as they operate.

4.2 Dynamic nature of Smart Transportation systems

Worse than the heterogeneity, the capabilities and context of devices in a Smart Transportation system can be changed over time (e.g varying CPU usage levels, moving vehicles). This highly dynamic nature requires the application execution to be able to recognize the changes and adapt to the changes. There is usually a trade-off between executing the adaptation in system-wide level and application-specific level. The earlier direction is more difficult to implement and the software architecture might get too comlex to be maintainable and/or portable. The later direction puts heavy burden on the developers, making the application codes complicated and error prone. Thus, finding the balance between these approaches remains a challenging task and will promise a very flexible execution platform and meaningful applications.

4.3 Large scale coordination of Smart Transportation System

Smart Transportation systems can consist of thousands of entities such as roadside units, street sensors or lamp posts, which are capable of providing computing resources. These entities are scattered widely over a large physical area such as a whole city. Thus, a Smart Transportation application while being distributed to such complex systems, has to monitor and control a huge number of devices. The complexity of such system suggests that simple programming techniques are simply insufficient to build the applications. Large scale characteristic of Smart Transportation systems has been addressed in many researches in WSN, however Smart Transportation systems usually do not inherit a common routing layer as found in most WSN, making the coordination among components difficult to realize.

5. STATE OF THE ARTS

While there are abundant researches that explore the promising of VANETs applications, most of these works centered around the experience of drivers, such as vehicular social network [14], public safety, crowd-sourced parking [4] or optimal driving routes; based on the lower layers such as routing and medium access control to deliver the application model; or limited at proposing the possible application scenarios. Application architectures that leverage the transportation infrastructure and vehicular data exist, but most of them exploit the Cloud Computing infrastructure so that the developed applications are executed in the Cloud while data are collected from the vehicles networks, such as ones found in [32, 15]. The extensive researches on the communication aspect of vehicular networks naturally lead to the bottom up approach in developing applications; however in our perspective, a common accessible application models and programming abstractions should be generally available for a typical developer to able to use.

5.1 Available application models and programming abstractions

Agent-based application model has been argued to be very suited for the domain of traffic and transportation systems due to their distributed nature [3]. Thus, there was a large body of researches that employ Agents in building applications in this domain [20]. Agents' execution is carried out by a distributed software platform called agent systems [3]. To allow Agents from different vendors to communicate with one another, researchers focused on the implementation of Agents that comply to FIPA standard [1] in regulating Agent-based technologies.

Due to the mobility of vehicles in VANETs, Mobile-Agent programming model is also used to develop applications [29] where Agents autonomously migrate from places to places while carrying with them data and computation processes. The authors in [17] proposed a Mobile Agent-based programming model for building vehicular social network application.

Beside Agents-based technology, The authors in [25] proposed a middleware layer that is build upon OSGi specification for dealing with the dynamic nature of VANET components. The applications is developed and executed in a

[1]http://www.fipa.org/

centralized computing infrastructure where data from vehicular and street sensors are collected by the middleware services. OSGi specification is relied on to create modular and evolving software stack for VANET applications.

Mobile Fog [13] is another effort in proposing a programming model for transportation application. Mobile Fog focused on application development in Fog-based Smart Transportation environment where components in Edge network such as vehicles or near Edge network such as Access network participate in the computation infrastructure.

5.2 Vehicular Cloud Models

The authors in [18] recommends a taxonomy of Cloud-based Vehicular Architecture including Vehicular Cloud where Vehicles themselves form a group of computing resources that can be rented out, Vehicles using Cloud where centralized Cloud infrastructure is used by vehicles and a Hybrid Cloud that combines these two. According to this taxonomy, current research into Intelligent Transportation Systems (ITS) is generally focused around exploiting resources from either vehicles or centralized computing infrastructure or both [7], thus following the Hybrid Cloud scheme. Smart Transport application architecture in our vision is more related to the Hybrid Cloud scheme, however, the Cloud resources are provided by both the transportation infrastructure and the traditional Cloud resources, forming a Fog Computing environment.

The authors in [12] proposed a vehicular computing infrastructure that follow the Hybrid Cloud model where a temporary Cloud formed by utilizing resources from vehicles compliments the traditional centralized Cloud resources. This approach has been embraced by several researches such as ones found in [1] and [28]. Hybrid Vehicular Cloud can be seen as the combination of Edge Computing infrastructure where vehicles operate and Cloud Computing. In this paper we argue that there exists another class of vehicular computing infrastructure that involves transportation infrastructure as well, these include but not limited to roadside units, communication base station and any resources between the Cloud and the Edge.

The author in [31] proposed an architecture called AutoNomos that signifies the importance of a decentralized data processing model in building scalable VANET applications. Applications are realized by aggregating and correlating "data items" from nearby cars together, thus forming a "Hovering Data Cloud" (HDC) where vehicles outside of a HDC can be informed by the aggregated and calculated data generated by a HDC.

The author in [9] proposed a Vehicular Fog architecture where Vehicular Contents which include vehicle-contributed sensory data and computing resources are used to deliver applications and services. The architecture has shown the important of Fog-based Vehicular Network in supporting autonomous vehicles due to the short networking delay and efficient content dissemination.

RT-STEAM [22] is a real-time decentralized event-based approach that aims at solving the real-time requirement of many VANET applications, such as dynamic traffic light coordination. RT-STEAM works based on the observation that event consumers are more likely interested in receiving the events from only the nearby producers. So event propagation could be done in a short proximity where QoS constraints are guaranteed. However, the work used a medium access control protocol in implementing the communication in a multi-hop ad-hoc network and the application model, therefore it does not proposed a programming abstraction that can be easily used by application developers.

6. CONCLUSIONS AND FUTURE DIRECTION

In this paper, we raised the need for more research efforts to be made in order to build and deploy a class of transportation applications that is widely distributed, following a Fog Computing model. It is perceived that while researches in Smart Transportation and VANETs in general usually focus on communication and architecture aspect, application models and programming abstractions do not receive much attention. This might due to the fact that the communication infrastructure for VANETs has not been realized significantly in real world. Nonetheless, with the advances in technologies in recent years, especially with the Internet of Things vision that is about to take off, crucial challenges in developing Smart Transportation applications should be well addressed.

To realize this future, several research directions are of interesting and necessary. Based on the challenges we addressed, it is obvious that large-scale coordination will be an interesting topic to be explored. For example, out of hundreds of vehicles passing by a street, an application might want to selectively collect data from or communicate with several ones that are available. Another application might want to coordinate a collection of traffic lights in a wide region in order to reroute the traffic. These scenarios require coordination of vehicles at large, which is a difficult task provided the dynamic nature and heterogeneity of participating vehicles. Applications deployment is another interesting aspect where it could be tricky to deploy a Smart Transportation application onto a widely distributed computing infrastructure. When it comes to mobile vehicles that host the computing resources, context-aware adaptation techniques are also important. This is because a significant part of Smart Transportation data comes from physical sources such as local flow speed of a street. When a vehicle moves around its running processes should be able to tell the physical context it's operating on to make correct decision.

In a nutshell, adequate researches into development of Smart Transportation applications will boost the capability of current transportation infrastructure and open new valuable applications and services with hard real-time requirements.

7. ACKNOWLEDGMENTS

The research is supported by NSERC IPS grant application ID 486401 co-funded by NSERC and Moovee Innovation Inc.

8. REFERENCES

[1] S. Bitam, A. Mellouk, and Z. Sherali. VANET-Cloud: A Generic Cloud Computing Model for Vehicular Ad Hoc Networks. *IEEE Wireless Communications*, (February):96–102, 2015.

[2] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli. Fog Computing and Its Role in the Internet of Things Characterization of Fog Computing. In *the first*

edition of the MCC workshop on Mobile cloud computing (MCC '12)*, pages 13–15, 2012.

[3] B. Chen and H. H. Cheng. A review of the applications of agent technology in traffic and transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 11(2):485–497, 2010.

[4] X. Chen, E. Santos-Neto, and M. Ripeanu. Smart parking by the coin. In *Proceedings of the Third ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, DIVANet '13, pages 109–114, New York, NY, USA, 2013. ACM.

[5] K. Dar, A. Taherkordi, R. Vitenberg, R. Rouvoy, and F. Eliassen. Adaptable service composition for very-large-scale Internet of Things systems. *Proceedings of the Workshop on Posters and Demos Track - PDT '11*, (DECEMBER):1–2, 2011.

[6] F. DeRemer and H. Kron. Programming-in-the large versus programming-in-the-small. In *International conference on Reliable software*, volume 10, pages 114–121, 1975.

[7] G. Dimitrakopoulos. Intelligent transportation systems based on internet-connected vehicles: Fundamental research areas and challenges. *ITS Telecommunications (ITST), 2011 11th International Conference on*, pages 145–151, 2011.

[8] Y. Gao, W. Hu, K. Ha, B. Amos, P. Pillai, and M. Satyanarayanan. Are Cloudlets Necessary ? (October), 2015.

[9] M. Gerla, E.-K. Lee, G. Pau, and U. Lee. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. *2014 IEEE World Forum on Internet of Things (WF-IoT)*, pages 241–246, 2014.

[10] N. K. Giang, M. Blackstock, R. Lea, and V. C. M. Leung. Developing IoT Applications in the Fog : a Distributed Dataflow Approach. 2015.

[11] M. Hajibaba and S. Gorgin. A Review on Modern Distributed Computing Paradigms : Cloud Computing , Jungle Computing. *Journal Of Computing & Information Technology*, 22(2):69–84, 2014.

[12] W. He, G. Yan, and L. D. Xu. Developing vehicular data cloud services in the IoT environment. *IEEE Transactions on Industrial Informatics*, 10(2):1587–1595, 2014.

[13] K. Hong, D. Lillethun, B. Ottenwälder, and B. Koldehofe. Mobile Fog : A Programming Model for Large âĂŞ Scale Applications on the Internet of Things. In *the second ACM SIGCOMM workshop on Mobile cloud computing (MCC '13)*, pages 15–20, 2013.

[14] X. Hu, V. C. M. Leung, K. G. Li, E. Kong, H. Zhang, N. S. Surendrakumar, and P. TalebiFard. Social Drive: A Crowdsourcing-based Vehicular Social Networking System for Green Transportation. In *Proceedings of the Third ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications*, DIVANet '13, pages 85–92, New York, NY, USA, 2013. ACM.

[15] X. Hu, L. Wang, Z. Sheng, P. TalebiFard, L. Zhou, J. Liu, and V. C. Leung. Towards a service centric contextualized vehicular cloud. In *Proceedings of the Fourth ACM International Symposium on Development and Analysis of Intelligent Vehicular Networks and Applications*, DIVANet '14, pages 73–80, New York, NY, USA, 2014. ACM.

[16] X. Hu, J. Zhao, D. Zhou, and V. Leung. A semantics-based multi-agent framework for vehicular social network development. *ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, pages 87–96, 2011.

[17] X. Hu, J. Zhao, D. Zhou, and V. Leung. A semantics-based multi-agent framework for vehicular social network development. *ACM International Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications (DIVANet)*, pages 87–96, 2011.

[18] R. Hussain, J. Son, H. Eun, S. Kim, and H. Oh. Rethinking Vehicular Communications: Merging VANET with cloud computing. *CloudCom 2012 - Proceedings: 2012 4th IEEE International Conference on Cloud Computing Technology and Science*, pages 606–609, 2012.

[19] K. Kai, W. Cong, and L. Tao. Fog computing for vehicular Ad-hoc networks: paradigms, scenarios, and issues. *The Journal of China Universities of Posts and Telecommunications*, 23(2):56–96, 2016.

[20] Z. Li, C. Chen, and K. Wang. Cloud computing for agent-based urban transportation systems. *IEEE Intelligent Systems*, 26(1):73–79, 2011.

[21] D. Lymberopoulos and A. Savvides. XYZ: A motion-enabled, power aware sensor node platform for distributed sensor network applications. In *2005 4th International Symposium on Information Processing in Sensor Networks, IPSN 2005*, volume 2005, pages 449–454, 2005.

[22] R. Meier, B. Hughes, R. Cunningham, and V. Cahill. Towards real-time middleware for applications of vehicular ad hoc networks. *Distributed Applications and Interoperable Systems*, pages 1–13, 2005.

[23] L. Mottola and G. P. Picco. Programming wireless sensor networks: fundamental concepts and state of the art. *ACM Computing Surveys*, 43(3):1–51, 2011.

[24] H. Moustafa and Y. Zhang. *Vehicular Networks: Techniques, Standards, and Applications*. Auerbach Publications, Boston, MA, USA, 1st edition, 2009.

[25] S. Nour, R. Negru, F. Xhafa, F. Pop, C. Dobre, and V. Cristea. Middleware for Data Sensing and Processing in VANETs. *Proceedings - 2011 International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2011*, 1:42–48, 2011.

[26] S. Sarkar, S. Chatterjee, and S. Misra. Assessment of the Suitability of Fog Computing in the Context of Internet of Things. *IEEE Transactions on Cloud Computing*, PP(99):1–1, 2015.

[27] R. C. Shah, S. Roy, S. Jain, and W. Brunette. Data MULEs: Modeling a three-tier architecture for sparse sensor networks. In *Proceedings of the 1st IEEE International Workshop on Sensor Network Protocols and Applications, SNPA 2003*, pages 30–41. Institute of Electrical and Electronics Engineers Inc., 2003.

[28] P. Trivedi, K. Deshmukh, and M. Shrivastava. Cloud Computing for Intelligent Transportation System. *International Journal of Soft Computing &*

Engineering, 2(3):568–572, 2012.

[29] O. Urra, S. Ilarri, T. Delot, and E. Mena. Mobile Agents in Vehicular Networks: Taking a First Ride. *Advances in Practical Applications of Agents and Multiagent Systems*, 70:119–124, 2010.

[30] L. M. Vaquero and L. Rodero-Merino. Finding your Way in the Fog. *ACM SIGCOMM Computer Communication Review*, 44(5):27–32, 2014.

[31] A. Wegener, H. Hellbrück, S. Fischer, B. Hendriks, C. Schmidt, and S. P. Fekete. Designing a Decentralized Traffic Information System - AutoNomos. *Proceedings of the 16. ITG/GI - Fachtagung Kommunikation in Verteilten Systemen (KiVS)*, pages 309–315, 2009.

[32] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen. Data-Driven Intelligent Transportation Systems: A Survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1624–1639, 2011.