ParkCar: A smart roadside parking application exploiting the mobile crowdsensing paradigm

Konstantina Banti
Department of Informatics and Telecommunications
Engineering School of Engineering,
University of Western Macedonia
Kozani, Greece
kbanti@uowm.gr

Malamati Louta
Department of Informatics and Telecommunications
Engineering School of Engineering,
University of Western Macedonia
Kozani, Greece
louta@uowm.gr

George Karetsos
Department of Computer Engineering
Technology Education Institute of Thessaly
Larissa, Greece
karetsos@teithessaly.gr

***Abstract*— In this paper, we design and implement a mobile crowdsensing smart roadside parking system, called ParkCar, exploiting the key elements that a mobile crowdsensing system (MCS) should possess. We present its architecture and basic operational characteristics, placing emphasis on the specific solutions adopted to respond to specific MCS open challenges, related mostly to efficient task assignment process, data quality and integrity, energy efficiency, security and privacy and incentive provisioning.**

***Keywords*— mobile crowd sensing; smart roadside parking; application design; architectures; challenges;**

I. INTRODUCTION

IN recent years, the trend of urbanization clearly stresses the need to focus on the design and deployment of smart cities, towards reducing CO_2 emissions and enhance energy savings [1]. Transportation constitutes one of the four pillars of a modern smart city [2]. In most countries, traffic congestion is widely recognized as a major problem, contributing significantly to environmental pollution, global warming, noise levels increments and depletion of fossil fuels, while it has negative effects on the overall quality of life [3, 4, 5, 6, 7]. For the reasons above, high volumes of traffic needs to be lessened. A significant contribution to urban congestion is attributed to drivers searching for available parking spaces [8]. It is generally recognized that finding parking spaces in a time-efficient manner constitutes a problem that should be efficiently and cost-effectively solved. To this end, smart parking applications are expected to significantly decrease the time needed to find a parking space and consequently reduce traffic congestion.

Most smart parking applications already proposed in related literature either do not consider roadside parking spaces and/or assume extensive deployment of sensors so as to detect the arrival/departure of vehicles [9,10,11,12]. Although this solution provides accurate information of available parking lots, its infrastructure deployment and maintenance is costly.

Recently, the proliferation of mobile devices in conjunction with the advancement in their communication, computing, storage and multi-modal sensing capabilities has given rise to a new sensing paradigm, the so called Mobile Crowd Sensing (MCS) [13]. MCS leverages on the power and wisdom of the crowd, exploiting human intelligence, ubiquity and mobility features, empowering the people to contribute data, sensed or generated, from their mobile devices. MCS has recently attracted the researchers' attention with applications ranging from environmental monitoring [14] to traffic planning [15], healthcare, public safety [16] to smart parking [6, 12]. However, a number of critical issues and challenges should be adequately addressed in order to MCS to achieve its full potentials [17].

Several smart parking applications exploiting the MCS paradigm have recently been proposed [6, 8, 12, 18]. Most of the solutions are based on a combination of sensor deployment and information collected from users who submit parking availability information. In this way, the application can propose a parking space which best suits the users' requirements, provisionally reserve it and subsequently provide guidance details. In this paper, we design and implement a mobile crowdsensing smart roadside parking system, called ParkCar. We present its basic operational characteristics and specific solutions which were adopted to tackle specific MCS open challenges.

The rest of the paper is structured as follows. Section II overviews smart parking systems already proposed. Section III highlights MCS characteristics and challenges that should be addressed when designing an MCS application. Section IV presents, in detail, the ParkCar application, focusing on how each design aspect is addressed. Finally, Section V concludes the paper and highlights our future plans.

II. RELATED WORK

The problem of searching for available parking spaces has been addressed by the scientific community in different ways.

Parking availability prediction [19], smart traffic management [20], architectures for parking management [21], etc. are examples of different approaches to mitigate the problems related with searching free parking spaces [22].

Several MCS paradigms have presented in smart parking solutions. In [18], the authors introduce an application which is built around a map view that shows the availability status of car parks, which by default follows the user's location. The application collects data with explicit user participation or monitors conspicuous facts that imply something about car parking availability. In [23], the authors present an application which prompts users to submit data about parking availability. They divide the parking area to lot zones to record user movement information. The application uses the accelerometer to register the motion so that can determine in which zone the user is located using the GPS and if the user is entering or leaving the parking lot. To reduce the influence of crowdsourcing data manipulation, expert data is also collected by an authority figure.

Furthermore, in [8] they authors have developed an application which collects parking availability information from drivers, who report their destination, their current location and car speed while driving and the parking availability on a certain street when they arrive. When a driver approaches his destination, the application searches for potential parking vacancies and informs the driver of the search result combined with a parking availability map. The devices can also collect geo-tagged sensor data automatically without drivers' intervention when the car is moving. Finally, the application rewards the participants according to their contribution level so that the incentives will be directly relevant to the application's goal. In [24] the authors present an application which approaches the problem of finding free parking spaces with a method utilizing primarily the activity recognition of two device states, in vehicle and on foot, to recognize if a user found to park and then he walks or if the parking area is full. This is combined with the location information. Any user can also manually input the status of a parking lot. They have tried to solve a number of challenges such as power consumption, incentives, and malicious users. In [25] the authors propose an application that detects available parking spots in cities using smart phones combined with ultrasonic sensing devices installed on vehicles.

III. Mobile Crowd Sensing Characteristics and Challenges

In this section, we briefly overview the most significant aspects and challenges that should be addressed in the context of MCS systems. MCS applications are based on data collection and analysis, applying appropriate data mining algorithms so as to identify spatio-temporal patterns, generate models and make predictions on physical or social phenomena being observed [13]. Data are collected both from the physical world (sensed data from mobile devices) as well as from online communities (mobile social network services related data) [26], which are often complementary, while humans are involved in the loop for data collection, processing, analysis and sharing [27]. MCS involves both implicit and explicit user participation in the data collection process with *opportunistic sensing* (which does not require users to actively act) and *participatory sensing* (that necessitates the active involvement of users) [13].

Numerous research challenges arise and should be adequately addressed. First, identifying and scheduling sensing tasks across multiple devices with diverse sensing capabilities and resource availabilities/limitations imposed is a quite complex issue to address. Second, taking into account the fact that different types of sensed data could be produced serving the same purpose, another challenging issue to solve is how to assure the quality level of the data collected, while minimizing the consumption of resources necessitated by the mobile devices. Third, the MCS systems should preserve the integrity of the collected data, considering also the case of potentially inaccurate / erroneous data provisioning. Thus, MCS systems should be highly adaptive and robust in order to adequately identify and address inadequate user participation, overcoming though the barriers imposed by obsolete, inconsistent and/or inaccurate data collections [28]. Fourth, considering the fact that privacy concerns are raised, MCS systems should incorporate a generic security framework so as to preserve the privacy of the individuals. Fifth, in order to guarantee adequate user participation, appropriate incentive mechanisms should be in place in order to promote user participation, retain their engagement to accurate data collection [29]. Finally, in order to cater for energy efficiency in MCS systems, it is of outmost importance to identify common data needs, allow for data sharing across different applications.

In [17], the authors critically surveyed MCS systems, models and architectures proposed in related research literature, analyzed their distinct features and discussed on their merits and weaknesses. They concluded that each MCS system is based on different assumptions, adopting different architectures/models, aiming to provide solutions to different sub-problems. They also included a list with the characteristics that MCS applications/architectures should possess, advancing context-awareness, self-adaptivity and advanced cognitive capabilities, while the potential interdependencies of MCS key features should be taken into account.

IV. ParkCar System

In this section, ParkCar, our proposed smart roadside parking system is presented in a detailed manner. ParkCar is based on the MCS paradigm, but it can be extended to combine with deployed sensory infrastructure. As a first step, the architecture of our crowd sensing system and the most important information about its operation are described and following on, the manner in which ParkCar addresses several MCS related challenges is discussed.

A. System Design

ParkCar follows the general architectural design of an MCS system, comprising the following main entities: a) the requestors, b) the workers and c) the platform. Thus, in ParkCar, the actors involved in the life cycle of a request are:

- CrowdSourcers-Requestors (CS-R): they issue requests pertinent to finding available parking spots in areas of interest, see the answers provided by CrowdWorkers to the platform (the subset pertinent to their issued request) and finally delete the requests submitted.
- CrowdWorkers-Participants (CW-P): that are assigned to several tasks pertinent to data collection, they are the main source of information and play major role in data collection. They can see the current tasks in the system, choose to participate or not in data collection and upload data about parking availability.
- Crowdsensing Platform (CP): the platform is the main communication link between CW-P and CS-R. Requests issued by CS-R and the related data are stored in the platform. Also, the platform stores and processes data provided from CW-Ps concerning issued requests and finally exports data to the appropriate CS-Rs and rewards workers' contribution.

At this point it should to be noted that a user can act as CS-R and CW-P at the same time. Thus, the system enables a user to issue a request and to participate in one or more tasks simultaneously.

ParkCar system architecture is graphically depicted in Fig. 1. It comprises the users' mobile application part (which currently has been implemented for mobile devices running Android OS) and the backend platform. The backend consists of a database and exploits the Google Cloud Messaging, which is used to notify users when certain conditions are met. Specifically, the system sends notifications, even in case the application is in the background, to:

- CW-Ps when they are selected to participate in a task.
- CS-Rs when there is an answer to their issued request.
- CS-Rs when issued requests are deleted.

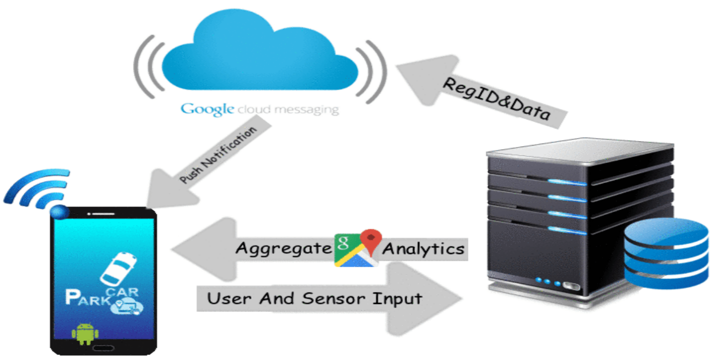

Figure 1. System Architecture

ParkCar has been designed following *context-aware* principles. Specifically, dynamic data pertinent to users' and devices' profile are exploited so as to improve the overall task assignment procedure. The location and the battery level of users' mobile devices are collected in specific time intervals, without the need for active involvement of the users, even when the ParkCar mobile application is in the background. In this sense, *opportunistic sensing* is performed. However, in order for a user to contribute to a task (i.e. provide parking availability information), his/her active participation is necessitated, following the *participatory sensing* paradigm. Thus, ParkCar involves both *explicit* and *implicit* user participation in the data collection process for improving the performance of the system. For addressing privacy concerns, ParkCar allows users to turn on and off the process of recording their locations at their own will.

B. *ParkCar application*

Once the user opens the ParkCar application, he/she is prompted to either sign up into the system by filling in requested information (i.e., username, password) or sign in if he/she is a registered user, filling in his/her username and his/her password. After logging in, the main page of the application appears with the following options (Fig. 2):

- **Account Information**: The user selects this option in order to see/modify his/her account information (all registered data as well as the overall reward the user has collected, contributing to task execution). It is noted that the user cannot modify the reward value.
- **Task Creation**: The user selects this option in order to create a new request concerning parking spot availability and submit it to the system, providing the requested information, such as city, address, and optionally other information that could be useful to CW-Ps for task execution (e.g., if a photo of the free parking is required), as shown in Fig. 3. The system enables the user to see his/her current location on the map, search an address and select a point of interest. The user is allowed to issue and submit more than one request at a time. Each submitted request terminates either by the requestor or after a pre-specified time period has passed.
- **Task Execution (Show Current Tasks)**: The user selects this option in order to browse the list of current tasks in the system that he/she can potentially contribute based on his/her recorded location. The user can select a task to view pertinent information (i.e., address of the point of interest and expiry time) and participate if he/she decides so. If the user chooses to participate, he/she can enter real-time data about parking availability, filling in information such as city, address, extra information requested by requestors, and/or take a picture of a free parking space. Finally, he/she can view on the map his/her current location, the location of a selected task and select the associated point via the map.
- **Parking Information (Show Answers)**: The user selects this option in order to see the answers returned by CW-Ps pertinent to his/her submitted tasks. The list of answers per request is displayed in an ascending order, showing first the responses with the minimum distance from the user's current location and the point of available parking specified in the answer. The user can see on the map the location of each provided answer, his/her own location and route directions (Fig. 4). Also, the user can open Google Maps via Park Car for extra navigation. Additionally,

the user can see the availability of spaces to different municipal parking places. Finally, the user may delete his task if he / she found a parking space.

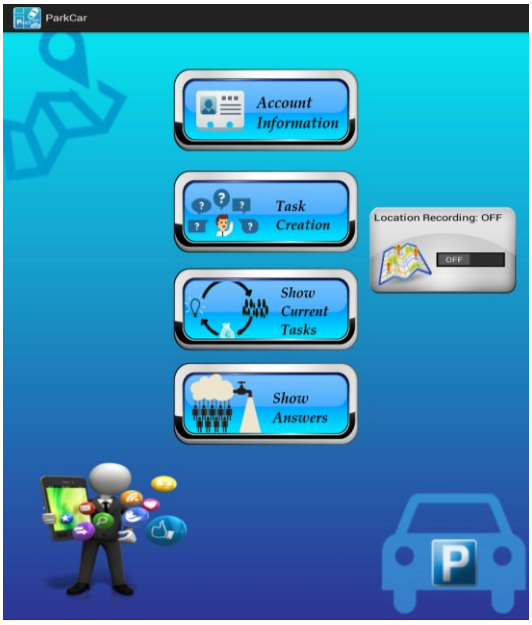

Figure 2. ParkCar Main Page

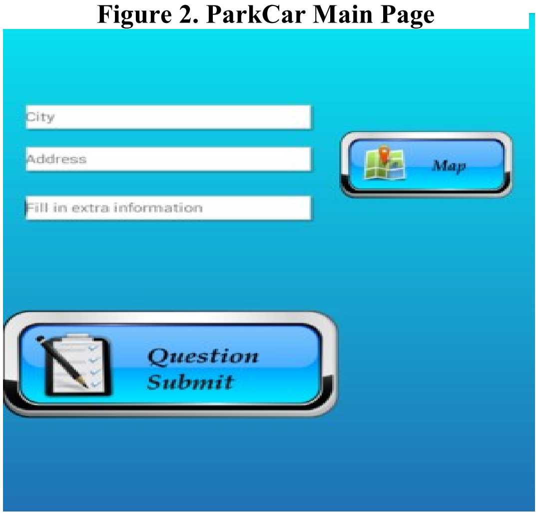

Figure 3. Task Creation

The life cycle of the task is illustrated in Fig. 5. When a user wants a free parking space, he/she creates a task providing details on the area of interest. Subsequently, the system notifies users that could contribute to the task. If a user wants to participate, he/she should submit the required data to the system within a time period. Following, the application sends a notification to the requestor about this new answer, which can be seen with map view. Finally, the system returns to CW-Ps a reward according to their contribution. The reward is calculated by the platform (platform-centric) and CW-Ps do not have control over the payment.

C. MCS Challenges Addressed

In the following we present the mechanisms implemented in ParkCar system in relation to several MCS challenges.

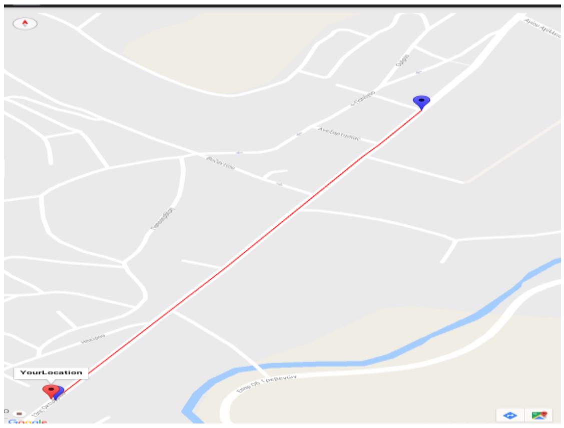

Figure 4. Show Route Directions

Task Assignment Process: In ParkCar, the overall task assignment process is context aware, is performed in an energy-efficient manner and follows the dynamic nature of users' presence within a specified area. Specifically, task assignment process is location dependent. Each time a CS-R creates a request, the system obtains the coordinates of the address specified and selects as potential CW-P users that are in close proximity to a pre-specified distance from the area of interest. By assigning crowdsensing tasks to the closest CW-Ps to the specified position, ParkCar endeavors to minimize the crowdsensing cost, minimize the time required for providing a response and maximize requestors' satisfaction, by finding available parking spaces near to the area of interest in a time efficient manner. Additionally, task allocation process takes into account the battery level of users' devices, so as to ensure that CW-Ps are enabled to provide accurate responses. Thus, the battery level of CW-Ps devices should exceed a certain threshold. At this point it should be noted that users receive task notifications in a push-based manner (not requiring users' intervention) according to their location (that is users that have deactivated location recording are not going to receive task notifications).

Task assignment is an iterative process. The area considered is split in a number of zones, considering different distances from the provided point of interest. The number of zones is dynamically formed per request; each zone comprises at least a pre-specified number of CW-Ps. Initially, the notification is sent to CW-Ps in the first zone. After a time period has elapsed, in case the task has not been deleted, the task assignment proceeds by sending a notification to CW-Ps in the next zone; otherwise, the task assignment process ends. This process takes place in consecutive steps until either the task is deleted or the current examined zone considers the whole area. In a nutshell, task assignment process is designed as an iterative process so as to consecutively assign tasks to the most appropriate CW-Ps, considering also the dynamic arrival of new CW-Ps in the considered area. This approach enhances ParkCar's Scalability, improves data quality and allows tasks to be distributed in a timely manner.

Data quality and accuracy: As already noted, the task assignment process aims at allocating the tasks to CW-Ps being closer to the point of interest so that CS-Rs acquire responses in a time efficient manner. Additionally, as soon as a CW-P submits an answer to the system, the system notifies the CS-R immediately. Moreover, taking into account the presence of selfish and/or malicious users that may provide erroneous data in purpose so as to disgrade the usefulness of the collected data and/or take advantage of certain situations and in order to ensure the validity and the integrity of the collected data, the system checks if the users are in close proximity (within few meters) from the position they indicate in their answer as being an available parking spot. If not, the answer is deemed as inaccurate is not returned to the CS-R. Finally, in order to cater for duplicate answer provisioning from the same CW-P indicating the same parking spot, the system checks the distance of the position indicated in the two answers; if it is less than a pre-defined distance, only one of the provided answers is deemed as valid. Finally, it should be noted that the ParkCar system providing also real-time availability of parking spaces in municipal parking areas thus improving the overall quality of data.

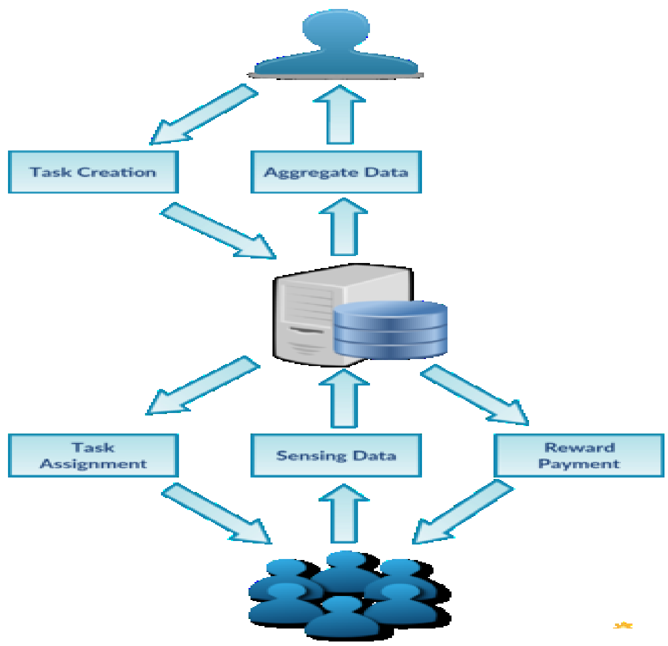

Figure 5. Task Life Cycle

Energy-Efficiency/ Resource Consumption Minimization: A service task assignment process monitors the users' location. Thus, users' devices send update messages to the system to keep their location information updated. Given that these update messages consititute an overhead, this process is performed every 10 minutes in order to minimize the number of update messages and to reduce energy consumption. Addtionally, updates to the system about battery levels of the users' devices are sent only when a change in the respective values is observed. This reduces the frequency and amount of data transferred from each device to the database. Aiming at resource consumption minimization, the system endeavors to involve in each task execution a minimum number of CW-Ps. Finally, the requested tasks and the respective collected answers are automatically deleted when the tasks have expired in order to reduce the data stored in the database and save resources.

Privacy and Security: MCS goals are accomplished on the basis of users' contributions. However, sensed data tagged with spatio-temporal information could be exploited in order to infer users' daily habits, routines and personal activities. Users would like to have access to a smart parking service, but, in general, they are reluctant in disclosing sensitive information (such as users' current location and points of interest). As privacy is user specific and should take into account users' willingness to share data, the ParkCar system allows them to control data collection and submission, enabling them to activate/deactivate whenever they wish location monitoring. Additionally, in our system only the last user's position is recorded, thus, mobility patterns cannot be inferred. Additionally, this information is binded to a user-id generated during user registration in the system. Thus, the user identity is not easily extracted. Moreover, for security reasons, ParkCar automatically releases users' connections after a time period has elapsed and notifies them to reconnect. In this way, ParkCar endeavors to guarantee that the system will not be accessed and used from other users in case devices are lost or stolen. Furthermore, once the user connection is released, location detection is not performed. Finally, as already pointed out, the system checks if a CW-P is found actually at the point indicated in his/her answer in order to ensure that the collected data is correct and the quality of the result is not undermined.

Incentives: In MCS systems, users may incur energy and computationally related resource consumption, monetary cost or are required to spend their time and put explicit effort in order to complete successfully their mission. In order for MCS systems to reach their full potentials, appropriate incentive mechanisms need to be in place so as to promote users cooperation and retain their engagement to accurate data collection or generation and sharing. In many cases, a monetary / virtual reward is provided to users for their participation to data collection.

In ParkCar, the reward returned to participants for their contribution to the system is in the form of virtual currency that may be used in order to gain access to other services. The total reward per request is predetermined; however, the reward per worker is estimated by the system upon the expiration of the task, taking into account a) the number of returned answers, b) the distance indicated in the answer as an available parking spot from the point of interest and c) the time the answer is submitted to the system. The answers indicating a close to the requested position available parking space, provided in a timely manner for a task that has collected a low number of answers will be higher rewarded. Specifically, the reward per CW-P for each task is estimated as follows. Let $budget_{task}$ denote the virtual amount the system will allocate to CW-Ps for each task and N the total number of answers returned to the system for a specific task. The maximum reward per returned answer ($budget_{ans}$) is calculated as $budget_{ans}=budget_{task}/N$. The reward returned for each answer is estimated as $reward_{ans} = w_1\ x\ rewardt_{ans}(t) + w_2\ x\ reward_{ans}(d) + w_3\ x\ reward_{ans}(N)$, taking into account the time t the answer is submitted to the system, the distance d of the indicated parking spot from the specified location of interest and the number of collected answers N. w_1, w_2 and w_3 are weighting factors and provide the relative significance of the three parameters to the estimation of the reward per answer. $reward_{ans}(f)$ *(f is equal to t, d, N)* is estimated as $reward_{ans}(f) = (1- l(f))\ x\ budget_{ans}$ where $l(f)$ is a factor calculated as $l(f)=(f-f_{min})/(f_{max}-f_{min})$

Time (t): indicates the time the answer of the CW-P is submitted to the system, t_{min} is the time of task creation and t_{max} is the time upon which the task will be deleted.

Distance (d): indicates the distance of the answer from the point of interest, d_{min} is the point of interest and d_{max} is the maximum distance from the specified location for which potential answers are taken into account.

Total amount of answers (N) per task. N is the total amount of collected answers per task, N_{min} is the minimum number of answers and N_{max} is the maximum number of answers that can be taken.

V. CONCLUSION

This paper demonstrates a smart roadside parking system, ParkCar, which exploits the MCS paradigm in order to collect parking availability information and let its users find a parking space. This application can be used by mobile devices running the Android OS. The authors aimed in addressing several MCS challenges. Next steps include experimentation in a real environment with an adequate number of users in order to identify and resolve possible design mishaps and address any disadvantages. Furthermore, specific performance indicators will be selected in order to objectively compare our application with similar ones from the literature.

REFERENCES

[1] R. Giffinger, "Smart City Concepts: Chances and Risks of Energy Efficient Urban Development", in International Conference on Smart Cities and Green ICT Systems, Springer, pp. 3-16, Portugal, 2015.

[2] European Commission Initiative on Smart Cities, 2010-2020, Available Online: http://setic.ec.europa.eu/set-plan-implementation/technology-roadmaps/european-initiative-smart-cities, accessed June 2017.

[3] E. Manley, T. Cheng, "Understanding Road Congestion as an Emergent Property of Traffic Networks", in Proc. of the 14th International Multiconference on Complexity, Informatics and Cybernetics, pp.109-114, Orlando, 2010.

[4] C. A. O'flaherty, "Traffic Planning Strategies", in Transport Planning and Traffic Engineering, Oxford, Butterworth-Heinemann, 2005.

[5] M. Papageorgiou et al, "Review of Road Traffic Control Strategies," in Proc. of Insitution of Electrical and Electronic Engineers, vol. 91, issue. 12, pp. 2043-2067, 2003.

[6] V. Coric, M. Gruteser, "Crowdsensing Maps of On-street Parking Spaces", in Proc. of the 9th Conference on Distributed Computing in Sensor Systems (DCOSS), IEEE, pp.115-122, USA, 2013.

[7] P. Masek et al, "A Harmonized Perspective on Transportation Management in Smart Cities: The Novel IoT-Driven Environment for Road Traffic Modeling", Sensors, vol.16, no.11, 2016.

[8] X. Chen, N. Liu, "Smart Parking by Mobile Crowdsensing", Internation Journal of Smart Home, vol.10, no.2, pp.219-234, 2016.

[9] SFPARK, Available Online: http://sfpark.org, accessed June 2017.

[10] Parkopedia, Available Online: https://en.parkopedia.com, accessed June 2017.

[11] S. Nawaz, C. Efstratiou, C. Mascolo, "ParkSense: A smartphone based sensing system for on-street parking," in Proc. of the 19th Annual International Conference on Mobile Computing & Networking, pp. 75–86, Florida, USA, 2013.

[12] R. Salpietro, L. Bedogni, M. Di Felice, L. Bononi, "Park Here! a smart parking system based on smartphones' embedded sensors and short range Communication Technologies", in Proc. of the 2015 IEEE 2nd World Forum on Internet of Things, pp. 18-23, Milan, Italy, 2015.

[13] R. Ganti, F. Ye, H. Lei, "Mobile crowdsensing: current state and future challenges", IEEE Communications Magazine, vol. 49, issue.11, pp. 32-39, 2011.

[14] C. Leonardi et al, "SecondNose: an air quality mobile crowdsensing system", in Proc. of the 8th Nordic Conference on Human-Computer Interaction, pp. 1051-1054, Helsinki, Finland, 2014.

[15] B. Pan, Y. Zheng, D. Wilkie, C. Shahabi, "Crowd sensing of traffic anomalies based on human mobility and social media", in Proc. of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems, pp. 344-353, Orlando, USA, 2013.

[16] C. Y. Lin, Y. Y. Su, H. H. Chu, "BikeTrack: Tracking Stolen Bikes through Everyday Mobile Phones and Participatory Sensing", in Proc. of the 2nd International Workshop on Sensing Applications on Mobile Phones, Seattle, USA, 2011.

[17] M. Louta, K. Banti, G. Karetsos, T. Lagkas, "Mobile crowd sensing architectural frameworks: A comprehensive survey", in 7th IEEE Internation Conference on Information, Intelligence, Systems & Applications (IISA), pp.1-7, Chalkidiki, Greece, 2016.

[18] J. Kopecký, J.Domingue, "ParkJamJAM: Crowdsourcing Parking Availability Information with Linked Data", Extented Semantic Web Conference, Springer, pp. 381-386, Berlin, Heidelberg, 2012.

[19] Y. Zheng, S. Rajasegarar, C. Leckie, "Parking availability prediction for sensor-enabled car parks in smart cities", in IEEE 10th International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), pp.1-6, Singapore, 2015.

[20] X. Sevillano, E. Marmol, V. Fernandez-Arguedas, "Towards smart traffic managemet systems: Vacant on-street parking spot detection based on video analytics", in 17th International Conference on Information Fusion (FUSION), pp.1-8, Salamanca, 2014.

[21] R. Barone, T. Giuffre, S. Siniscalchi, M. Morgano, G. Tesoriere, "Architecture for parking management in smart cities", Intelligent Transport Systems, IET, vol. 8, no.5, pp.445-452, 2014.

[22] F. Villanueva, D. Villa, M. Santofimia, J. Barba, JC Lopez, "Crowdsensing smart city parking monitoring" In 2nd IEE World Forum on Internet of Things (WF-IoT), IEEE, pp.751-756, Milan, Italy, 2015.

[23] J. Villalobos, B. Kifle, D. Riley, JUQ Torrero, "Crowdsourcing Automobile Parking Availability Sensing Using Mobile Phones", in UWM Undergraduate Research Symposium, pp.1-7, University of Wisconsin-Milwaukee, USA, 2015.

[24] M. Rinne, S. Törmä, "Mobile crowdsensing of parking space using geofencing and activity recognition", in 10th ITS European Congress, pp. 16-19, Helsinki, Finland, 2014.

[25] Suhas Mathur et al, "Parknet: drive-by sensing or road-side parking statistics", in Proc. of the 8th International Conference on Mobile Systems, Applications and Services, ACM, pp.123-136, USA, 2010.

[26] B. Guo, Z. Yu, D. Zhang, X. Zhou, "From Participatory Sensing to Mobile Crowd Sensing", in Proc. of the 12th IEEE International Conference on Pervasive Computing and Communications Workshops (PERCOM Workshop), pp. 593 - 598, Budapest, Hungary, 2014.

[27] H. Ma, D. Zhao, P. Yuan, "Opportunities in mobile crowd sensing", IEEE Communications Magazine, vol. 52, issue. 8, pp. 29-35, 2014.

[28] P. Y. Chen et al, "When crowdsourcing meets mobile sensing: a social network perspective", IEEE Communications Magazine, vol. 53, issue 10, pp. 157-163, 2015.

[29] T. Giannetsos, S. Gisdakis, P. Papadimitratos, "Trustworthy People-Centric Sensing: Privacy, Security and User Incentives Road Map", in Proc. of the 13th Annual Mediterranean Workshop on Ad Hoc Networking, pp.39-46, Piran, Slovenia, 2014.