People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

Kasdi Merbah University of Ouargla
Faculty of New Technologies of Information and Communication
Department of Computer Science and Information Technology

Doctoral Thesis

Thesis submitted in partial fulfillment of the requirements for the degree of PhD 3rd cycle in computer science.

Option :

Networking and Telecommunications

Towards an efficient mobile and social sensing via Internet of Vehicles

Presented by

M^r Yesin Sahraoui

Jury members

President:	*Akram Zineddine Boukhamla*	M.C.A, University of Ouargla, Algeria
Examiner:	*Amine Khaldi*	M.C.A, University of Ouargla, Algeria
Examiner:	*Said Bachir*	M.C.A, University of Ouargla, Algeria
Examiner:	*Younes Guellouma*	M.C.A, University of Laghouat, Algeria
Advisor:	*Chaker Abdelaziz Kerrache*	M.C.A, University of Laghouat, Algeria
Co-advisor:	*Ahmed Korichi*	Professor, University of Ouargla, Algeria

-2022/2023-

For my parents.
For my sisters and brother.
For my little familly.

Acknowledgements

First and foremost, I praise due to Allah, for every thing.

Apart from my very little efforts , the success of this thesis depends largely on the encouragement and guidelines of many others.

I take this opportunity to show my greatest appreciation to my thesis supervisor Dr. Chaker Abdelaziz Kerrache who kept me going when I wanted to throw in the towel. I can't say thank you enough for your tremendous help, motivation, encouragement every time, and guidance, without your high-level supervision this thesis would not have materialized

I gratefully acknowledge the deepest gratitude to my co-supervisor Pr. Ahmed Korichi, for his motivation, support, encouragement and assistance, which made me come into contact with high-level researchers such as Dr. Chaker.

Besides my supervisor and Co-supervisor, I would like to thank the rest of my thesis committee: Dr. Akram Zineddine Boukhamla, Dr. Amine khaldi ,Dr. Said Bachir, and Dr. Younes Guellouma, for their insightful comments for the improvement of this study from various perspectives, and generously spent precious time in giving the guidance and suggestion.

Special thanks are extended to Dr. Marica Amadeo, Dr. Anna Maria Vegni from Italy for the cooperation during research. I am grateful for their constant support and help.

Last but not least, it would be difficult to find adequate words to thanks my family, my friends for their support all time, It would be impossible to list all names.

May God bless you all.

Abstract

The ubiquity of smart phones and the popularity of social media heralds an era of mobile and social sensing and detection. This latter becomes a paradigm for collecting observations of physical phenomena, either directly from human observers or by crowdsourcing data measurement tasks using sensors in smart phones or various other portable devices (for example, Google Glass, Apple Watch and Fitbit devices). Thus, it is a massive amount of sensory and / or scattering data collection where humans can act as carriers of sensors (for example, carriers of GPS sensors that share location data), operators sensors (eg, take pictures and videos with smartphones), or as source of information (eg, share comments on social media). This emerging field faces new challenges in collecting, disseminating, merging and mining data, privacy and security, to name a few. On the other hand, as vehicles evolve from simple means of transport to intelligent entities with new detection and communication capabilities, they become active members of a smart city. The Internet of Vehicles (IoV) consists of vehicles that communicate with one another and with public networks via V2V (vehicle-to-vehicle), V2I (infrastructure vehicle) and V2P (pedestrian vehicle), which enables both the collection and sharing in real time of critical information on the state of the road network. The Social Internet of Things (SIoT) introduces social relations between objects, creating a social network where participants are not humans, but intelligent objects. In this thesis, we present new solutions to limit the spread of some widespread viral diseases and deal with some concerns. Based on the IoV paradigm, we conduct our research with a focus on high cost, limited geographic coverage, lack of privacy, and data quality. Specifically, the main contributions are, i) to improve sensing time and ensure large-scale geographic coverage, at a low cost, we propose a remote sensing based-IoV scheme to limit the spread of Covid-19; ii) a deep-learning-based crowd management scheme, namely DeepDist is introduced to control physical distancing violations among pedestrians and safety distancing among vehicles; iii) to enable monitoring of hard-to-reach areas thus ensure full coverage, and delete replicate data in sensing reports, we propose GUAVA, an efficient framework that uses ground vehicles assisted by flying vehicles for sensing tasks; iv) to increase identify exposed people to the virus in a simple and easy way, we propose TraceMe, a contact tracing framework, that relies on Mobile and Wireless Networks (MWNs) and Online Social Networks(ONS). The results obtained based on simulation, show the efficiency of the proposed schemes in terms of QoS metrics, in addition, to the ability to sense large-scale areas in a short time, which contributes to making timely decisions.

Keywords: Internet of Vehicles, Social sensing, Mobile Crowdsensing, Internet of Things, Online Social Networks.

Résumé

L'omniprésence des smartphones et la popularité des médias sociaux annoncent une ère de détection mobile et sociale. La détection mobile et sociale devient un paradigme pour la collecte d'observations de phénomènes physiques, soit directement auprès d'observateurs humains, soit en crowdsourcing des tâches de mesure de données à l'aide de capteurs des smartphones ou divers autres appareils portables (par exemple,Google Glass, Apple Watch et appareils Fitbit). Il s'agit donc d'une collecte massive de données sensorielles et/ou diffusantes où les humains peuvent agir comme porteurs de capteurs (par exemple, porteurs de capteurs GPS qui partagent des données de localisation), opérateurs de capteurs (par exemple, prendre des photos et des vidéos avec des smartphones), ou en tant que source d'information (par exemple, partager des commentaires sur les réseaux sociaux). Ce domaine émergent fait face à de nouveaux défis dans la collecte, la diffusion, la fusion et l'extraction de données, confidentialité et sécurité, pour en nommer quelques uns. D'autre part, avec l'évolution des véhicules, de simples moyens de transport à des entités intelligentes dotées de nouvelles capacités de détection et de communication, ils deviennent des membres actifs d'une ville intelligente. L'Internet des véhicules (IoV) est constitué de véhicules qui communiquent entre eux et avec les réseaux publics via V2V (véhicule à véhicule), V2I (véhicule d'infrastructure) et V2P (véhicule piéton), ce qui permet à la fois la collecte et le partage en temps réel d'information critique sur l'état du réseau routier. L'Internet social des objets (SIoT) introduit des relations sociales entre les objets, créant un réseau social où les participants ne sont pas des humains, mais des objets intelligents. Dans cette thèse, nous présentons de nouvelles solutions pour limiter la propagation de certaines maladies virales répandues et répondons à certaines préoccupations. En se basant sur le paradigme d'IoV, nous menons nos recherches en mettant l'accent sur le coût élevé, la couverture géographique limitée, le manque de confidentialité et la qualité des données. Plus précisément, les principales contributions sont les suivantes : i) pour améliorer le temps de détection et assurer une couverture géographique à grande échelle, à faible coût, nous proposons un schéma d'IoV basé sur la télédétection pour limiter la propagation du Covid-19 ; ii) un système de gestion des foules basé sur l'apprentissage profond, à savoir DeepDist, est introduit pour contrôler les violations de la Distanciation physique entre les piétons et la distance de sécurité entre les véhicules ; iii) pour permettre la surveillance des zones difficiles d'accès afin d'assurer une couverture complète et de supprimer les données en double dans les rapports de détection, nous proposons GUAVA, un framework efficace qui utilise des véhicules terrestres assistés par des véhicules volants pour les tâches de détection ; iv) pour augmenter l'identification des personnes exposées au virus de manière simple et facile, nous proposons TraceMe, un framework de traçage des contacts, qui s'appuie sur les réseaux mobiles et sans fil (MWN) et les réseaux sociaux en ligne (ONS). Les résultats obtenus sur la base de la simulation montrent l'efficacité des schémas proposés en terme de QoS, en plus de la capacité de détecter des zones à grande échelle en peu de temps, ce qui contribue à prendre des décisions en temps opportun.

Mots-clés : Internet des véhicules, détection sociale, crowdsensing mobile, Internet des objets, Réseaux sociaux en ligne.

ملخص

إن انتشار الهواتف الذكية في كل مكان وشعبية وسائل التواصل الاجتماعي يبشر بعصر من الكشف المحمول والاجتماعي. حيث اصبح الاكتشاف المحمول والاجتماعي نموذجًا لجمع ملاحظات الظواهر الفيزيائية ، إما مباشرة عن طريق مراقبين بشريين أو عن طريق التعهيد الجماعي لمهام قياس البيانات باستخدام أجهزة الاستشعار في الهواتف الذكية أو العديد من الاجهزة المحمولة الاخرى(على سبيل المثال: نظارة غوغل ، ساعة ابل و أجهزةFibit)، وبالتالي ، جمع قدر هائل من البيانات الحسية و / أو المتشتتة حيث يمكن للبشر العمل كناقل لأجهزة الاستشعار)على سبيل المثال ، ناقل لأجهزة الاستشعار GPS لمشاركة بيانات الموقع(، أو مشغل مستشعر (على سبيل المثال ، التقاط الصور ومقاطع الفيديو باستخدام الهواتف الذكية) ، أو كمصدر للمعلومات (على سبيل المثال ، مشاركة التعليقات على وسائل التواصل الاجتماعي). يواجه هذا المجال الناشئ تحديات جديدة في جمع ونشر ودمج وتعدين البيانات ؛ الموثوقية والخصوصية والأمن، على سبيل المثال لا الحصر . من ناحية أخرى ، مع تطور المركبات من وسائل نقل بسيطة إلى كيانات ذكية ذات قدرات اكتشاف واتصالات جديدة ، فأصبحت أعضاء نشطة في المدينة الذكية. تتكون إنترنت المركبات (IoV) من مركبات قادرة على التواصل مع بعضها البعض ومع الشبكات العامة عبر V2V (مركبة إلى مركبة) ، V2I (مركبة الى بنية تحتية) و V2P (مركبة الى مشاة) ، والتي تتيح كل من جمع ومشاركة المعلومات الهامة في الوقت الحقيقي عن حالة شبكة الطرق. يقدم إنترنت الأشياء الاجتماعي (SIoT) العلاقات الاجتماعية بين الكيانات، مما يؤدي إلى إنشاء شبكة اجتماعية حيث لا يكون المشاركون بشرًا ، بل كائنات ذكية. نقدم في هذه الرسالة حلولاً جديدة للحد من انتشار بعض الأمراض الفيروسية المنتشرة والتعامل مع بعض المخاوف. استنادًا إلى نموذج IoV ، نجري بحثنا مع التركيز على التكلفة العالية والتغطية الجغرافية المحدودة ونقص الخصوصية وجودة البيانات. على وجه التحديد ، المساهمات الرئيسية هي: 1) تحسين وقت الاستشعار وضمان تغطية جغرافية واسعة النطاق ، بتكلفة منخفضة ، نقترح مخطط IoV القائم على الاستشعار عن بعد للحد من انتشار Covid-19 ؛ 2) نظام إدارة الحشود القائم على التعلم العميق ، وهو برنامج DeepDist الذي تم تقديمه للتحكم في انتهاكات التباعد الجسدي بين المشاة والمسافة الآمنة بين المركبات ؛ 3) لتمكين مراقبة المناطق التي يصعب الوصول إليها ، وبالتالي ضمان التغطية الكاملة ، وحذف البيانات المكررة في تقارير الاستشعار ، نقترح GUAVA ، وهو إطار فعال يستخدم المركبات الأرضية بمساعدة المركبات الطائرة لمهام الاستشعار. 4) لزيادة التعرف على الأشخاص المعرضين للفيروس بطريقة بسيطة وسهلة ، نقترح TraceMe، وهو إطار عمل لتتبع جهات الاتصال ، يعتمد على شبكات المحمول واللاسلكية (MWN) والشبكات الاجتماعية عبر الإنترنت (OSN)النتائج التي تم الحصول عليها بالاعتماد على المحاكاة ، تظهر كفاءة المخططات المقترحة من حيث مقاييس جودة الخدمة ، بالإضافة إلى القدرة على استشعار مناطق واسعة النطاق في وقت قصير ، مما

يساهم في اتخاذ القرارات في الوقت المناسب.

الكلمات المفتاحية: إنترنت المركبات ، الاستشعار الاجتماعي ، الاستشعار الجماعي المتنقل ، إنترنت الأشياء، الشبكات الاجتماعية عبر الإنترنت.

Contents

List of Figures **xiii**

List of Tables **xv**

List of Abbreviations **xvi**

I Introduction and context 1

1 Introduction 3

1.1 Context of the thesis . . . 3
1.2 Motivation and Contributions . . . 4
1.3 Thesis outline . . . 5

2 Internet of Vehicles: An overview 7

2.1 Introduction . . . 7
2.2 Definition and characteristics . . . 8
2.2.1 What is IoV? . . . 8
2.2.2 Distinguishing features of IoVs . . . 8
2.3 Vehicular communications types . . . 9
2.3.1 Basic communications . . . 9
2.3.2 Extended communication: . . . 10
2.4 Layered IoV architecture and protocol stack . . . 11
Protocol stack . . . 12
2.5 Applications of IoV . . . 14
2.6 Security aspect of IoV . . . 15
2.7 Summary . . . 17

3 Mobile crowdsensing **18**
3.1 Introduction 18
3.2 Crowdsensing 19
3.2.1 MCS process 21
3.2.2 Incentives for participants 22
a) Monetary incentives 22
b) Non-monetary incentives 22
3.2.3 MCS applications 23
3.2.4 MCS challenges 24
3.3 IoV-based crowdsensing 26
3.4 Mobile crowdsensing and pandemics 27
3.5 Conclusion 28

II Contributions **29**

4 IoV-based Crowdsensing to control respiratory viral diseases outbreaks **31**
4.1 Introduction 31
4.2 Related Work 32
4.2.1 IoT for healthcare and epidemics detection 32
4.2.2 IoV-based remote sensing 33
4.3 System Design 33
4.3.1 Prevention Strategies 35
4.4 Performance Evaluation 37
4.4.1 Dataset 37
4.4.2 Simulation and results 39
A. QoS metrics: 39
B. Accuracy measures: 42
4.5 Summary 43

5 DeepDist:Deep learning-based physical distancing violations in IoV ecosystem. **44**
5.1 Introduction 44
5.2 Related Work 45
5.2.1 Measuring Physical Distance 45
5.2.2 Object Detection and Recognition 46
5.3 Preliminary: An overview of Faster R-CNN algorithm 47
5.3.1 Architecture 47
5.3.2 Training and Loss Functions 48

5.4 DeepDist Framework . . . 49
5.5 Evaluation . . . 51
5.6 Discussion . . . 52
5.7 Concluding Remarks . . . 54

6 GUAVA: Ground vehicles assisted UAV crowdsensing frAmework to combat the spread of covid-19. 55
6.1 Introduction . . . 55
6.2 Existing Work . . . 57
6.2.1 Vehicle-based crowdsensing . . . 57
6.2.2 UAV-based crowdsensing . . . 58
6.3 The GUAVA Framework . . . 59
6.3.1 System model . . . 60
6.3.2 Remote sensing process . . . 63
6.3.3 Face recognition process . . . 66
6.4 Evaluation . . . 68
6.4.1 Simulation setup . . . 68
6.4.2 Results . . . 70
6.4.3 QoS metrics evaluation . . . 72
6.5 Discussion . . . 73
6.6 Concluding Remarks . . . 74

7 TraceMe: Contact Tracing Framework based on OSN for Early Prevention of Infectious Disease Outbreaks. 76
7.1 Introduction . . . 76
7.2 Related Work . . . 77
7.2.1 IoT based-CT apps . . . 77
7.2.2 Applying SNA for CT . . . 79
7.3 TraceMe Model . . . 80
7.3.1 MWN-based contact tracing . . . 80
7.3.2 OSN-based contact tracing . . . 81
7.4 Evaluation . . . 84
7.4.1 Description of Dataset . . . 84
7.4.2 Social network analysis . . . 84
7.4.3 A theoretical Forecasting using the SEIR model . . . 86
7.5 Concluding Remarks . . . 89

8 Conclusion, Future Work and Publications. 90
8.1 Contributions summary . . . 90
8.2 Future Research Opportunities . . . 92
8.3 Publication list . . . 92

List of Figures

2.1 A seven-layered architecture of IoV adopted from [6]. 13

3.1 A typical architecture of MCS adapted from [12]. 20
3.2 Classification of incentives. 23

4.1 The proposed design for identifying suspected cases of the COVID-19 virus. 35
4.2 Exploiting the proposed architecture for prevention purposes. . . . 36
4.3 Suspected cases of viral diseases heatmap. 38
4.4 QoS metrics ((a) E2E delay; (b) PDR; (c) Throughput; (d) Jitter) . 41
4.5 Graph to represent false negative and false positive. 43

5.1 The general architecture of faster R-CNN. 48
5.2 The general architecture our proposal approach. 50
5.3 Extended DENM to support safety distance violation alerts. 51
5.4 Average PDR variation according to vehicles density. 52
5.5 Mean E2E delay variation according to vehicles density. 52
5.6 Precision/recall (PR) curve for Faster R-CNN. 53

6.1 General design of our GUAVA framework. 61
6.2 Example about UAVs monitoring trajectory. 62
6.3 GaussianFace architecture to identify matching faces in GUAVA framework. 68
6.4 E2E (network) delay vs. the UAVs/GVs density. 71
6.5 Total E2E delay vs. the UAVs/GVs density. 71
6.6 PDR variation vs. UAVs/GVs density. 72
6.7 Throughput variation vs. UAVs/GVs density. 72
6.8 Mean jitter variation vs. UAVs/GVs density. 72
6.9 GUAVA additional overhead compared to BUCST. 73
6.10 GUAVA area monitoring convergence compared to BUCST. 73

7.1 Overview of TraceMe approach based on MWNs and OSNs. 80
7.2 Cliques-based community detection in TraceMe to identify and notify close contacts. 82
7.3 Users at high risk of infection (*blue nodes*) when varying the number of infected users (*red nodes*). 85
7.4 The diagram of the SEIR compartmental model 87
7.5 Modeling pandemic outbreak with respect to time by varying the rate the use of ONS. 88

List of Tables

1 Some attacks and threats in IoV. 16

1 Synthetic dataset of suspected cases. 37
2 Parameter Setting in the Simulation. 39

1 Simulation parameters. 69
2 Error rate result. 70

1 Users at risk according to the number of infected users. 85

List of Abbreviations

DDos . . . . . Distributed Denial of Service.

DENM . . . . Decentralized Environmental Notification Message.

DSRC . . . . . Dedicated Short Range Communication.

HOG . . . . . . Histogram of oriented gradients.

IoTs . . . . . . Internet of Things.

ITS . . . . Intelligent Transportation System.

GIDs . . . . . . Global IDentifications.

MCS . . . . . . Mobile crowdsensing.

MWNs . . . . Mobile and Wirelless Networks.

OSN . . . . . . Online Social Network.

QoE . . . . . . quality of experience.

QoI . . . . Quality of information.

RFID . . . . . Radio-Frequency Identification.

SIoVs . . . . . Social Internet of Vehicles.

SNA . . . . . . Social Network Analysis.

SVM . . . . . . Support Vector Machines.

V2X . . . . . . Vehicle-to-Everything.

VANETs . . . Vehicular Ad-hoc Networks.

VCNs . . . . . Vehicular Communication Networks.

VCS . . . . . . vehicular crowdsensing.

WAVE . . . . . Wireless Access in Vehicular Environments.

Part I

Introduction and context

"The greatest challenge to any thinker is stating the problem in a way that will allow a solution."

— Bertrand Russell

Chapter 1

Introduction

Contents

1.1 Context of the thesis **3**
1.2 Motivation and Contributions **4**
1.3 Thesis outline **5**

1.1 Context of the thesis

In the last few years, Internet of Thing (IoT) technology has revolutionized our world, and shaped a major breakthrough towards the future, for enhancing our daily lives, and allowed everyone to be connected to the internet, for uploading and downloading data, without or with human limited intervention [1]. The IoT smart sensors have emerged as one of the most promising technologies on our days, they are interacting with the physical world, where considered as the eyes, ears, and nose of IoT applications. The latter allowed users to share their personal thoughts, observations about the surrounding environment for the public interest, which stimulated the emergence of mobile and social sensing frameworks.

The appearance of mobile and social sensing frameworks, and their impact on the ways to get massive and varied amounts of data, more quickly and accurately than ever. The built-in sensors of mobile devices (e.g., camera, microphone, gyroscope, etc.) collect and share information then using analytics strategy, to make reliable predictions for better decisions-making, backed with a visual display of data such as highly precise maps, charts, and diagrams.

1.2 Motivation and Contributions

The conventional approaches of mobile and social sensing face various challenges, including power consumption associated with sensing, processing, and analysis capabilities, high sensing cost, limited communication ability, limited geographical coverage, and low storage space, data reliability issue, security, and privacy sensitivity.

To address and mitigate the aforementioned challenges, and motivated by the proliferation of the Internet of Vehicles (IoV), which emerged as a promising and exciting paradigm, supporting advances made in the IoT ecosystem, and stretched to cloud computing technologies. Shedding light on this emerging and powerful strategy, which revolutionized the world of crowdsensing and caused great changes in the way to collect data for various purposes, such as healthcare, environment, transportation, etc, leveraging the mobility feature and the existence of various heterogeneous networks to support needs.

The objective of this thesis is to design a new and efficient mobile and social sensing framework based on IoV paradigm, leveraging its smart devices and computational capabilities, thus, offering timely and accurate delivery of data, and ensuring wider geographic coverage, which will be then used for different applications including disaster management, crowd management, and intelligent traffic lights, to improve decision-making.

To fulfill the aforementioned objectives, we design a suite of schemes based on IoV framework, aim to enhance Mobile Crowd Sensing(MCS) campaigns. Specifically, the main contributions are as following:

- In the first contribution, a new way of mobile and social sensing has been proposed to mitigate and contain the pandemic. Specifically, we design a framework to contain COVID-19 outbreak using IoV, provide a *real-time data collection* (including body temperature, breathing rate and positioning information) of pedestrians, and ensure *full city-scale coverage*, *without incentive budget*, which allows localizing the affected areas, thus, adopt preventive strategies and make a better decision so quickly. Evaluations with a synthetic dataset of Annaba city, show the great performance of our design in terms of QoS metrics.

- The second contribution is a kind of crowd management in smart cities, combining IoV and deep learning technologies to detect physical distancing violation, and then display a real-time notification for practicing physical distancing, with an accurate prediction and low cost, using Faster-CNN algorithm, it called: DeepDist(A Deep-Learning-Based IoV Framework for Real-Time Objects and Distance Violation Detection), where each vehicle is equipped with a switching camera system through thermal and vision imaging.

- The third contribution implement a smart sensing framework namely GUAVA, that relies on the coordination between UAVs and ground vehicles (GVs), where the sensing campaigns to detect humans' vital parameters in hard-to-reach areas are assigned to UAVs. The duplicate data are identified by GaussianFace algorithm by discovering matching faces, In addition to the adoption of wireless charging techniques through nearby GVs.

- The fourth contribution implement a new solution based on OSNs and social network analysis, in addition to the traditional physical proximity detection relies on MWNs, and applying the notion of cliques-based OSN for the detection of communities exposed to the disease.

1.3 Thesis outline

This dissertation shows, across eight chapters, how IoV paradigm can be leveraged to increase the performance of crowdsensing campaigns, in addition to OSNs. The rest of this thesis is organized as follows:

- **Chapter 2:** presents an overview of IoV technology, focused on its communication types, layered-based architecture, and protocol stack, with mention the three major types of its applications.
- **Chapter 3:** introduces conventional MCS, lists some of the main challenges it faces, and then explains how IoV is changing the way of MCS, by highlighting its use in intervention situations during pandemics.
- **Chapter 4** present our work for remote sensing based on IoV. Specifically, we design a framework to contain the pandemic outbreak using IoV, provide a real-time data collection strategy, and ensure full city-scale coverage. Extensive simulations are carried out and discussed showing the suitability of this solution in terms of QoS metrics.
- **Chapter 5** introducing our deepDist work, introducing a new design based on IoV, to detect physical distancing violation using deep learning technique.
- **Chapter 6** addresses the coverage of some specific geographic areas such as hard-to-reach ones, to ensure full coverage, through the use of UAVs in coordination with the Ground vehicles, enhanced by an artificial intelligence algorithm to recognize faces and exclude duplicate data.
- **Chapter 7** introducing our TraceMe framework, a new and easy to use design, leveraging Cliques-based OSN coupled with WSNs to detect exposed people to the Covid-19.
- **Chapter 8** concludes this thesis, where we give a brief summary of the whole work, with some directions for the future work.

"Being present is being connected to All Things."

— S. Kelley Harrell

Chapter 2

Internet of Vehicles: An overview

Contents

2.1 Introduction 7
2.2 Definition and characteristics 8
2.3 Vehicular communications types 9
2.4 Layered IoV architecture and protocol stack 11
2.5 Applications of IoV 14
2.6 Security aspect of IoV 15
2.7 Summary 17

2.1 Introduction

The great development in the Smart Transportation System (ITS), led to a great interaction between people, vehicles, and infrastructures, thus the emergence of the Vehicular Ad-hoc NETworks (VANETs) paradigm, to assist driver, improve safety, and ensure comfortable on the roads. However, some of VANET's limitations, including its small-scale deployment, technical and business-related limitations, pushed it for more advanced and capabilities expansion. Moreover, the Internet of Vehicles (IoV) pattern has become the other face of the smart city, to improve the effectiveness of their management system, by exploiting the Internet of Things (IoT) technology. where the next generation of connected vehicles integrated heterogeneous wireless access networks (e.g. DSRC/WAVE, WIFI, 4G/5G LTE, etc.), which allowed to work with different applications including vehicle-oriented, driver-oriented, and passengers-oriented applications, hence, providing multiple services and more collaboration. while IoV promises exciting features, it still faces several challenges, especially cybersecurity issues.

2.2 Definition and characteristics

2.2.1 What is IoV?

The Internet of Vehicles (IoVs) also known as *Vehicular Communication Networks (VCNs)*, C. Jiacheng et al. in [2], defined IoVs to be the key pillar for Intelligent Transport System (ITS) aims to offer many different services, ranging from security purpose to internet browsing. J. Zhang and B. K. Letaief in [3]. considered that IoV is the evolution of conventional VANETs in Smart City, leveraging Cloud Computing technology, and IoT paradigm, with built-in machine learning strategy, to extend its capabilities and provides smarter services including traffic management, vehicle insurance, logistics, etc. W. Xu et al. in [4]. defined IoV as a way of supplying both the conventional VANETs and the data center with powerful computational devices and high-capacity storage, to provide real-time reliable services.

We can conclude that the IoV is a distributed network of vehicles, sensors, and mobile IoT devices, connected to the global network, with the goal of expanding the capabilities of VANET and deliver real-time services.

Furthermore, in order to add the social dimension, and to allow different interactions between passengers, pedestrians, vehicles, and drivers, the *Social Internet of Vehicles (SIoV)* paradigm is appeared, which considered as the next step of IoV [5], to enhance the social relationship and get a real-time massive data of the surrounding environment.

2.2.2 Distinguishing features of IoVs

Among some of the most important characteristics that distinguish IoV from conventional VANET, we mention:

- **Connectivity:** in IoV all vehicles stay Always Connected through multi-interface GIDs, using different wireless access networks, with various communication ranges and under different bandwidth connections, including DSRC/WAVE, WIFI, and 4G/5G LTE. Unlike VANET which cannot provide continuous connectivity.

- **Compatibility and scalability:** With the advanced and diversity of IoT technology, IoV has integrated many heterogeneous devices and implemented it with various communication models and protocols, which made it very smooth and highly scalable, and therefore, make it communicate with different devices, and solve the compatibility issue found in VANET, due to the nature of its architecture.

- **Computing and analysis power:** VANET only has limited applications, due to the lack of processing and analysis capacity to collect and process big data, therefore, IoV has used other technologies such as cloud computing, to improve its computational and storage capability and to collect information on a large scale, thus obtaining accurate decisions in real-time.

- **Collaborative architecture:** vehicles in IoV pattern can work in a collaborative manner, as offloading the information into a single data center allowed to save time and reduce cost.

2.3 Vehicular communications types

2.3.1 Basic communications

- **Vehicle-to-Vehicle (V2V):** vehicles are able to communicate with each other, using dedicated short-range communications (DSRC) to exchange crucial information including speed, direction, and location, using wireless channels and multi-hop techniques, with the help of various sensors it provides 360-degree awareness of its surroundings, and make drivers receive warning information, such as blind spots and dead angles warning, and crash avoidance, etc.

- **Vehicle-to-Roadside/Infrastructure (V2R/V2I):** this communication enables vehicles to share useful information with RoadSide Unit (RSU), to improve road safety conditions, find parking zones, and avoid traffic jam situation, which in turn broadcast it using the high bandwidth with nearby vehicles and RSUs. V2R/V2I also uses the DSRC standard for communications.

2.3.2 Extended communication:

The rising number of autonomous and connected vehicles, contributed to the emergence of a new concept, namely Vehicle-to-Everything ecosystem (V2X), which includes the following subcategories:

- **Vehicle-to-Pedestrian (V2P):** this communication means that V2X system can detect and share information directly with pedestrians through their mobile devices including smartphones, tablets, and wearable devices, which helps alert pedestrians and drivers of a potential accident at the right time and location.

- **Vehicle-to-Network (V2N):** enable a vehicle to communicate with other mobile devices such as smartphones, tablets, and wearable devices, leveraging cellular networks and the internet.

- **Vehicle-to-Sensors (V2S):** smart vehicle is relying on a great number of onboard sensors, which makes it become aware of its surroundings, using in-vehicle data acquisition devices such as LiDAR, to detect obstacles and make drivers more aware.

- **Vehicle-to-Broadband cloud (V2B):** this communication means that vehicles can communicate via wireless broadband mechanisms such as 4G / 5G LTE, which allows to share and exchange data with the cloud. This type of communication will be useful for active driver assistance and vehicle tracking.

2.4 Layered IoV architecture and protocol stack

Different designs and layered architectural patterns of IoV proposed in the literature, based on layers capabilities, including information-network performance, security and communication technologies, etc, among them: Three-layered IoV model architecture,four-layered IoV model architecture, Five-layered IoV model architecture, and here we present the seven-layered architecture based on [6], as shown in Figure, which consists of:

- **User interaction layer:** This layer relies on providing unconventional interfaces for direct interaction with the driver without distracting him, depending on the severity of the situation. For example, when a traffic accident occurs in front of the vehicle, flash lighting appears on the dashboard with a sound warning system.

- **Data acquisition layer:** This layer contains all in-vehicle sensing devices such as built-in cameras, GPS trackers, etc, wearable devices, smartphones, and roadside units (RSUs). These sensors are responsible for collecting data from the surrounding environment, determining vehicles' positions, driving patterns, etc. The vehicle can also have a radio-frequency identification (RFID) tag, which contains a vehicular global ID (GID) or cyber license plate.

- **Data filtering and pre-processing layer:** Because sensors in IoV disseminate data as it is, without filtering, this layer is designed for pre-processing the data, filtering process to keep only relevant information, and eliminating redundancy, using artificial intelligence techniques, thus improve the performance in disseminating information, and avoid overloading the network.

- **Communication layer:** this layer is designed to determine the suitable network (e.g. 4G/5G LTE, satellite, WiMax, WiFi, Bluetooth, etch), using some intelligent selection approaches including Multiple Attributes Decision Making (MADM) algorithms [7], and fuzzy algorithms [8] , according to various parameters, such as level of Qos, information relevance, privacy and security over available networks, that offers seamless and full connectivity to transmit information.

- **Control and management layer:** This layer is designed to synchronizing the diverse network service providers in the IoV environment, and managing the data exchanged between different services, and applied some techniques such as packet inspection, traffic engineering, and traffic management policies, to improve the management of sensed information.

- **Business layer:** This layer is designed for processing a massive amount of data, analyzing, storing, processing the information, using different cloud computing types including IaaS (Infrastructure-as-a-Service), PaaS (Platform-as-a-Service), SaaS (Software-as-a-Service), and others. It contains a set of data visualization tools including graphs, charts, and tables, to be exploited in the business model predictions, proposes new strategies, and helps to make decisions.

- **Security layer:** This layer communicates with all other layers, and deployed multiple security measures including authentication, integrity, non-repudiation and confidentiality, access control, availability, authenticity & trustworthiness, etc, to stand up against various cyberattacks and ensure network security in the IoV environment.

Protocol stack

The design of protocols for VANET is currently an active research area for communication, and its effect of interaction between different layers, Nevertheless, IoVs increased the scope of communication in VANETs, and

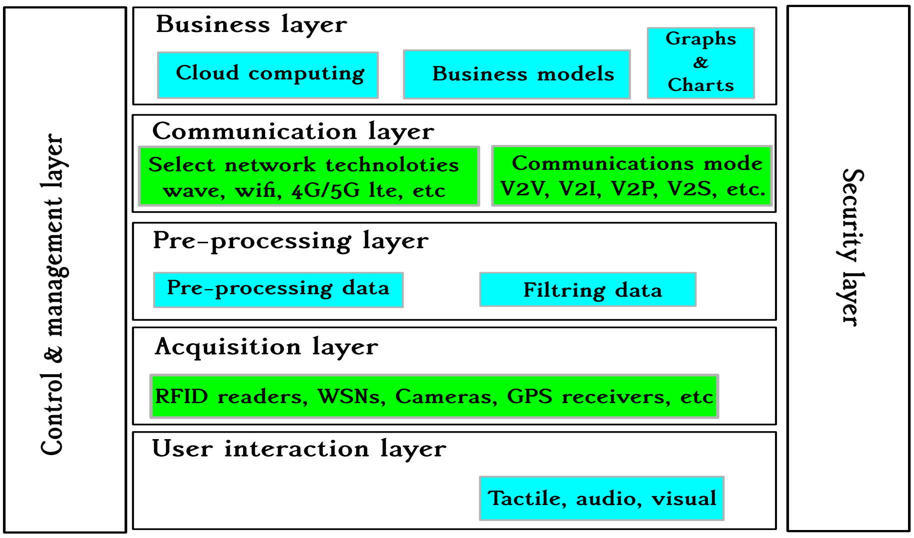


Figure 2.1: A seven-layered architecture of IoV adopted from [6].

provided various forms of interaction, including with humans, surrounding environment, cloud, etc, by designing further protocols according to different research projects, to provides highly reliable and real-time communication for non-safety (frequencies range from 5.855 MHz to 5.875 MHz) and safety applications (frequencies range from 5.875 MHz to 5.905 MHz).

The interaction layer provides the main interfaces for the drivers/passengers to interact with the different applications using voice, visual display, and touch. The acquisition layer includes protocols for intra-vehicle communications such as RFID, Bluetooth, WAVE (Wireless Access in Vehicular Environments) that follows IEEE 1609.x standards, WiFi, and others to communicate with the external environment of the vehicle such as 802.11p-based Dedicated Short Range Communication (DSRC), 4G/5G LTE, WIMAX, IEEE 802.15.4, and low-power WAN. Many protocols have been designed in the pre-processing layer such as Extensible Messaging and Presence Protocol (XMPP), lightweight local automation protocol (LLAP), and Constrained Application Protocol (CoAP), to facilitate exchanges of structured information among nodes in the IoV environment. The protocols in the communication layer ensure routing and transportation, they can be considered complementary of IP and TCP/UDP protocols, among these, we mention Routing Protocols for Low Power and Lossy Networks (RPL) and IPv6 LoW Power wireless Area Networks. The management layer contains protocols for the service management process, such as WAVE-1609.6 service, and Technical Report 069 (TR-069) for remote management. The business layer contains Analytics & BigData protocols in the cloud for more analysis and decision-making. The security layer provides a set of measures and protocols to secure access to wireless networks.

2.5 Applications of IoV

The emergence of the large number of applications related to smart cities and connected vehicles greatly helped in improving transportation services and make them more sustainable, and provide positive effects on our daily life, by decreasing accidents and pollution degree and increasing awareness around us, however, the main applications of IoV are divided as follows:

- **Safety applications:** means the applications that focus on reducing crashes between vehicles, avoiding dangerous road conditions, and other safety measures, using early warning, thus reducing accidents rate and related burdens and save a lot of lives.

- **Traffic efficiency and management:** these applications enable traffic management in an automated manner without human intervention, by sending periodic query messages by vehicles to inquire about traffic, and thus obtain crucial information, such as traffic jam situations, closing lanes due to construction works, accidents ahead, the information enables it to change lanes and thus makes reduced traffic and congestion.

- **Comfort and infotainment applications:** designed to make the ride delightful and easier, and help to enhance the driving experience with less effort, it includes parking space detection, advanced navigation systems, online gaming and videos streaming through the dashboard, etc.

- **Healthcare and Environment Monitoring applications:** these applications contribute to public health by sending information related to human health, captured by sensors, to health care centers to take the necessary measures, such as sending personal health information(PHI) of an injured in real-time for early diagnostic including blood pressure, body temperature, pulse rate, hence, gives pre-medical treatment, these applications can also monitor the environment around us, including air/ noise pollution, climate change, and natural disasters.

- **Vehicle diagnostics and maintenance applications:** the smart car contains a self-diagnostic system, which is able to locate the potential malfunction and repair it, if possible, display a warning message, or stop and contact the nearest vehicle maintenance center, before accidents can occur, for instance, tire-pressure monitoring using sensors mounted in the wheels, and display a light warning when they are high/low on air, using Tire Pressure Monitoring System (TPMS).

2.6 Security aspect of IoV

The rapid evolution of the IoVs made driving easier and led to optimal management of smart cities, but on others hand, many different risks emerged, which make the IoV ecosystem vulnerable to various types of active and passive attacks [9], as summarized in the table below.

It is very important to ensure security measures in the IoV ecosystem to avoid fatalities, at the network level or cloud computation, therefore, many researchers proposed some schemes to solve these issues and protect the network. The following are the security requirements of IoV:

- **Authentication:** to be involved in the data exchange, the vehicle must first authenticate for sending and forwarding messages, using its unique ID.

- **Availability:** the information must be available at the right time, otherwise it will be worthless.

Table 1: Some attacks and threats in IoV.

Type of attack	**Short description**
∘ Sybil attack	-the attacker uses multiple identities
∘ Denial of Service(DoS) attack ∘ Distributed DoS (DDoS) attack	-the attacker causes congestion in the network, and making services and resources inaccessible. -variant of DoS, It attacks on a distributed manner.
∘ Blackhole attack ∘ Grayhole attack	-intermediate vehicles either drop the forwarded packets or refuse to participate in the communication. -variant of blackhole attack where vehicles partially drop packets through selective method
∘ Worm attack	-two or more malicious vehicles create a private network and force packets to follow it.
∘ GPS Spoofing attack	the attacker gives a fake location.
∘ Social attack	-the attacker send emotional messages on the network.
∘ Illusion attack	-the attacker correctly authenticate and generate false information.
∘ Replay attack	-the attacker re-transmit the forwarded messages many times.

- **Confidentiality:** setting some restrictions to limit access to sensitive information, which gives it some privacy.

- **Data integrity:** indicates that the data has not been altered, the digital signature can be used to validate the integrity of the information [10].

- **Non-repudiation:** relies on encryption mechanisms such as digital signatures, to ensure that the sender/receiver will not deny that the message was sent.

- **Privacy:** it indicates the rules taken to limit the sharing of personal information through unauthorized access.

2.7 Summary

We have present in this chapter the concept of IoV that inspired from IoT, as the vehicle became more intelligent, autonomous, and connected, with offering safe and fast transit, by making use of its different forms of communication, we discussed a seven-layered model architecture that allows seamless communication with heterogeneous devices, and sharing data in IoV ecosystem while highlighting some security issues.

The fleet of connected vehicles equipped with different sensors to collect data and provide a new opportunity to show a new type of mobile crowd, this is what we will see in the next chapter.

"A crowd exists so long as it has an unattained goal."

— Elias Canetti

Chapter 3

Mobile crowdsensing

Contents

3.1 Introduction 18
3.2 Crowdsensing 19
3.3 IoV-based crowdsensing 26
3.4 Mobile crowdsensing and pandemics 27
3.5 Conclusion 28

3.1 Introduction

The world today is witnessing an unprecedented proliferation of applications for gathering data, as the exchange of information has become a feature of modern societies, and plays a central role in our lives. Whoever possesses accurate and timely information, will possess all factors of power, influence, and control the behavior of people, wars, and economies. Some sensitive data needs security and adequate protection from loss, damage, and theft, and then apply filters, and organize data in a structured way, hence, delivering information to Stakeholders and especially Decision Makers. There are different data-gathering methods, by specifying the information to be collected and the timeframe for that, and then choosing the appropriate method, among the most significant and efficient approaches in data collection the Mobile crowdsensing.

3.2 Crowdsensing

Crowdsensing (also known as *Mobile Crowdsensing*) refers to the process of gathering data associated with diverse phenomena in specific areas [11], using sensing and computing devices(eg., smartphones, smartwatches, tablets, etc.), carried by participant users, and then extract information, with the aim of map, estimate, predict any event of common interest. Crowdsensing generally exists in two types according to user involvement:

(1) Participatory crowdsensing where users will be recruited to participate voluntarily, and they have awareness of the sensing process, using their mobile devices for both collecting and sending sensing data; ***(2) Opportunistic crowdsensing*** where the information is autonomous acquired in the background, with low user's involvement, and in some cases, even without awareness or prior permission of users.

A typical MCS campaign usually consists of three entities: participants, dedicated application, and back-end platform, as shown in Figure 3.1.

- *Participants:* considered as the main part of the crowdsensing that deliver different types of information through their smart devices including traffic information, social information (e.g. location data, behavior, and relationship), health information (e.g. vital sign data, medical test results), environment information (e.g. weather status, disaster prevention), etc. Typically a participant must have an account for acceding and participating in sensing tasks.

- *Dedicated applications:* the sensing applications are downloaded and deployed on mobile devices, using Application Delivery platforms (e.g. Apple AppStore, Google Play Store, etc.), which providing an easy-to-handle interface, and allow to have a useful way of gathering information depending on the application or participants themselves.

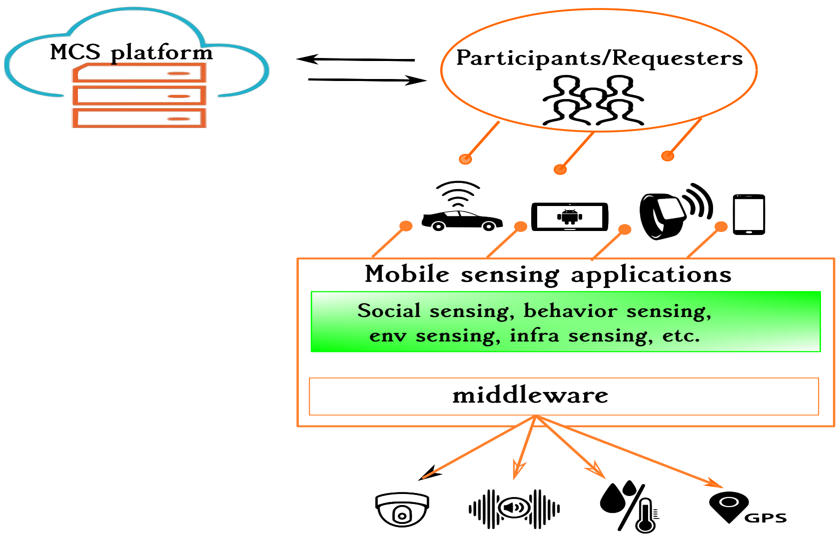


Figure 3.1: A typical architecture of MCS adapted from [12].

- *Back-end platform:* comprised data processing and analytics, where most computations are performed including filtering, mining, storage, and interpretation of data using visual display, depending on charts and graphs. A reward-based incentive strategy can be adopted to reward users for their contributions to crowdsensing, thus, helping to increase the performance and participation rate. These platforms offer several methods for evaluating a user's reputation and predicting the trustworthiness of sensed data, based on other users' opinions and past behavior.

Among the most important features that distinguish the MCS, we mention:

– **Openness:** MCS is an open framework which means that each entity either malicious or not, can participate using its sensing devices, as well it can join or leave the sensing campaigns freely, which makes this framework vulnerable to several security attacks, and increase The challenges of managing the whole system.

– **Unreliability:** as a result of the openness framework, unreliability occurs when injecting untrusted information by malicious nodes, which affects the overall result through the untrustworthiness sensed data.

- **Mobility:** the dynamic and mobile nature of MCS improves its sensing and coverage ability, however, it will run into issues of managing multiple entities.
- **Data massiveness:** MCS is capable of collecting a huge amount of diverse data in a short time, by leveraging its mobility feature, but this diversity increases the difficulty of processing and thus decreases the accuracy.
- **Network Heterogeneity:** the collected data can be shared using different heterogeneous networks such as the coexistence of Wi-Fi with LTE or other types of networks, which provides permanent communication, but this leads to the emergence of new patterns of security threats.

3.2.1 MCS process

Typically, the process of MCS can be divided into four stages: task creation, task allocation, task execution and crowd data aggregation.

- **Task creation:** the MCS organizers create the sensing tasks focusing on simplifying them as well as providing an understandable and easy-to-use interface for participants who have deployed later the app on their smart devices.The main issue in this stage is how to improve the efficiency of MCS task creation[11].
- **Task allocation:** once the MCS organizers have created the task, it must be assigned to the participants using different recruiting algorithms such as a reverse auction. MCS task assignment is considered a critical issue by striving to ensure low-cost coupled with high-quality [13] and finding enough participants.
- **Task execution:** this stage involves the use of mobile devices build-in sensors such as cameras, GPS trackers, etc. Once receiving the MCS task by a participant, he will immediately start sensing, computing, and then uploading the data, taking into account the Spatio-temporal constraint. The main issue in this stage is how to conserve energy as long as possible [14].

- **Crowd data aggregation:** in this stage, for helping good decision-making, and reply to the query of requesters, with the efficient manner and minimum delay, all reported data of the crowd are aggregated, using data analytic techniques. The raw data need pre-processing (e.g filtering, transforming, and cleaning, etc), and then mapping out and creating a visual representation to understand the phenomena.

3.2.2 Incentives for participants

Incentives considered as strong motivators of behavior, especially in MCS environments. Requesters offer incentives to boost participation rates and encourage users to contribute to the collection and sharing of data. Typically, incentives can be monetary or non-monetary rewards (e.g. services, entertainment, and social incentives) [15], as shown in the Figure 3.2:

a) Monetary incentives

represented in real or virtual currencies, such as bitcoin, This type of incentive can be static, it is determined in advance and then does not change over time, while dynamic one is based on the lowest wage rate at which a user will accept participation, depending on some factors, such as type of data, location, time and number of participants [16], also monetary incentives can be platform-centric, where prices are set to achieve maximum profits for the platform, or user-centric, where the prices are set by the participants [17].

b) Non-monetary incentives

this kind of incentives can be classified into three types, consisting of the following: i) *Entertainment incentives*, this type relies on incentivizing users for crowdsensing, in ways that make them feel pleasure (e.g, designing games for both entertaining and MCS). For example, CachedSensing [18] is designed as a game and environment monitoring application. ii) *Social incentives*, the need to belong and socialize with people and undertake ethical initiatives, socially desirable behaviors, and altruism, to create a nice social image, good reputation and gaining social acceptance [19],

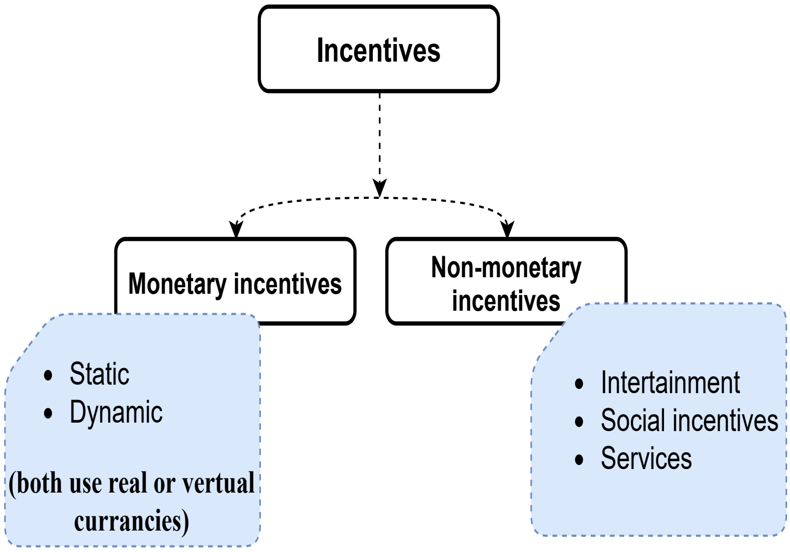


Figure 3.2: Classification of incentives.

motivate people to participate in sensing tasks, especially when their behavior is observable by others. For example, STAR [20] a crowdsensing system that leverages social trust for synergistic marriage and mutual benefit. iii) *Service incentives*, give participants services for their contributions in sensing campaigns, for instance, getting free calling and texting, free internet data, discounts, etc.

3.2.3 MCS applications

Mobile Crowdsensing applications can be divided into four main domains based upon multiple sensing phenomena [21], including social, healthcare, infrastructure, and environmental phenomena:

- **Environmental applications:** these applications became very prevalent, using the sensing capabilities of the crowds, to promote sustainable awareness about the environmental issues, using their smartphone and other mobile devices to collect and share environmental information, such as air/noise pollution, atmospheric conditions, volcanic eruptions, wildlife habitats monitoring, etc, for forecasting and early warning of natural disasters, to make rapid decisions. For instance, Third-Eye [22], is a mobile application to monitor air pollution in cities, depending on deep learning technology.

- **Infrastructure applications:** it includes all applications related to the Intelligent Transport Systems (ITS), especially in urban areas, to enhance traffic safety and avoid crashes, help improve urban traffic flow and reduce congestion, real-time parking management, etc. For instance, CRAPER [23] is a crowdsensing framework to evaluate road conditions using RESTful interface.

- **Health and well-being apps:** These applications enable the collection and sharing of health-related data, whether emotionally or physically, to help to make informed decisions, speed in healthcare delivery, as well as reducing overall health costs, and improving quality of life and well-being. For example, vital signs monitoring in real-time via wearable sensors and smartphones [24], Allergymap application for management of allergic diseases [25], MyFitnessPal [26] is a necessary well-being application to track calories and monitor their diet, especially for diabetics.

- **Social applications:** these applications provide context-aware services, push people to cultivate their relationships, focusing on issues that impact their everyday habits, help to make the world happier and easier, by improving the users' concentration and more active, such as activities/services suggestion, recommendation of parking areas, travel assistance, etc. For example, ASPIRE [27] is a smart and lightweight parking recommendation system for vehicular networks.

3.2.4 MCS challenges

MCS is facing major challenges regarding resources (battery, processing capability, etc), incentives, QoI, security, privacy, and trustworthiness of data.

– **Resource limitations:** is the critical constraint for users to participate in crowdsensing, during sensing, processing, and uploading operations, sensing applications consume a considerable amount of device resources (e.g. energy, computation power, bandwidth, and memory), thus, negatively affects the crowdsensing campaigns, and makes users alienate participation. A lot of solutions have been adopted to cope with limited resources, for example, GSM combined with low-cost GPS [28], can be used to determine a location, to avoid greatly draining batteries, while sacrificing a few accuracy [29]. Also, the data collection phase can involve *deduplication* strategies, to eliminate the redundant data, and increase storage space, using a real-time deduplication processing or post-deduplication processing strategy.

- **Security, privacy and trust:** the openness of the MCS platform to facilitate sensing campaigns, as well as the aspect of mobility behavior, made it vulnerable to serious security threats. When participants launch their apps to collect different kinds of information, they are unaware of data that will be shared, which is considered as invasion of privacy, as it can be used to predict behavioral patterns of individuals, and it will be difficult to be controlled it during collecting, processing, and uploading phase, for example, when sharing some personal information such as time and location-related information, that could be exploited by burglars to steal things. Many researchers have resorted to addressing this issue of exposing personal information by hiding it adopting some solutions including encryption, cloaking, anonymous and facial blurring. Data trust is one of the biggest challenges that MCS faces, the users' observations can be false and contradictory with each other. Accordingly, verify the validity and trustworthiness of data reports, some solutions have been adopted to prevent unreliable participants, such as their historical behaviors, abilities, and availability [30], etc.

- **Budget constraints:** monetary incentives are among the most used techniques as rewards, compared to other incentives such as gamification or providing services, crowdsensing providers give incentives to encourage users to participate in crowdsensing campaigns, and people by nature tended to offer reliable data for significant rewards. Among widespread mechanisms, we find reverse auction [31], to increase participants' enthusiasm.

- **Quality of information (QoI):** also known as data quality, it was defined by Bisdikian et al. in [32] for wireless sensor networks as the degree of usefulness or relevance of sensed data applied in a context to understanding a phenomenon. QoI problem is considered among the main issues facing MCS, that may be affected by an uncontrollable movement path, it arises through duplicate and unstructured data, incomplete and suitability of data, or hard-to-access data. So, for collecting and delivering information for each sensing campaign, data must be passed by a QoI estimation phase [33], and subjecting it to all constraints or properties set by the crowdsensing providers in terms of accuracy, relevancy, completeness, usability, and timeliness, called QoI constraints, to make judgments about sensed data.

3.3 IoV-based crowdsensing

In addition to traditional approaches for crowdsensing, that depend on mobile devices, MCS can also include smart vehicles combined with the Internet of Things (IoT) devices, namely IoV frameworks, which provide a new opportunity for mobile sensing, where each vehicle is equipped with a set of embedded multi-modal sensors, and onboard units (OBUs), with powerful processing capabilities, and benefit from ubiquitous Internet connectivity and its greater mobility, for gathering a large amount of data, such as, traffic monitoring, air/noise pollution, predicting disasters, crime control, etc. The conventional vehicular ad-hoc networks(VANETs) rely on simple communication, namely vehicle-to-vehicle (V2V) or vehicle-to-Infrastructure (V2I) communications, which are not cloud-connected, and not willing to deliver such services, because of some application that requires low latency, and without discontinuous connectivity, which negatively affects the quality of experience (QoE), and therefore, quality of service (QoS)[34].

In contrast to conventional approaches of mobile sensing, *vehicular crowdsensing(VCS)* provides a cost-effective solution, without energy constraints, and holds enormous sensing, computing, and communications capabilities, which allows to collect of information and deliver services in real-time, enables stakeholders to make accurate decisions instantaneously, and leveraging the benefits of its high mobility to perform large-scale sensing campaigns, and ensure full coverage in a short time. VCS has also benefited from network infrastructure to improve their performance, and increase the potential for storing, analyzing, and sending huge amounts of data, by integrating cloud computing technologies, or bring these services closer at the edge of the network, using *Fog-enabled vehicle crowd-sensing (FEVC)* [35], to increase its performance, and help to achieve more flexibility and reliability in the data processing, while maintaining the feature of *local relevance*[34], which means that The data acquisition depends on spatial scope and utility based on lifetime. For example, safety applications require information in real-time, otherwise, the acquired data is useless.

3.4 Mobile crowdsensing and pandemics

The fast outbreak of some pandemics such as infectious respiratory disease including COVID-19, H1N1, SARS, etc, and its significant impact on the public health of the population, with high economic burdens due to total lockdown and social distancing measures taken by governments, especially, developing countries. All of these things led to thinking about a new approach to monitor and track affected areas using MCS, as it is considered one of the effective methods and powerful solution in terms of i) Providing remote sensing technology to avoid direct contact with infected objects. ii) Improving the method of collecting information based on collaboration and achieving high-quality data across incentives. ii) allow achieving full Geo-coverage in a brief time. iii) real-time data-driven decision-making, by identifying areas containing pandemics and taking different preventive measures. In addition, all the tremendous information that is collected is used by epidemiologists to understand the phenomenon or understand the pandemic itself and strive to find final solutions and avoid catastrophic health outcomes. Many research efforts focused on the use of MCS to address pandemics outbreaks. The authors in [36–38], present new frameworks based on mobile sensors, to fight the spread of the COVID-19 pandemic, also, the human being can behave like a sensor, namely social sensing, by sharing his observations using online social networks (OSN). For example, the authors in [39], design a new framework namely WATERSensing, to prevent and evaluate epidemic disease that affects water, using social networks

3.5 Conclusion

The plethora of heterogeneous wireless sensors, increased the importance of hybrid-sensing and crowdsensing in different domains, using vehicle-based-sensing, smartphone-sensing, etc, and leveraging modern communication networks, that aims at understanding and interpreting the various phenomena, and allow to predict and avoid some situations, such as increasing traffic safety. Besides, these new sensing patterns promoted the smart city concept, and make life easier and less stressful.

This chapter gave an overview of MCS and its application domains, especially in the pandemic era, outlining the prominent issues that MCS faces to be tackled, and highlighting one of its classes, which is vehicular-based sensing.

Part II

Contributions

““Every wheel wish to be the wheel of a car, and not of just another vehicle.””

— Amit Kalantri

Chapter 4

IoV-based Crowdsensing to control respiratory viral diseases outbreaks

Contents

4.1 Introduction **31**
4.2 Related Work **32**
4.3 System Design **33**
4.4 Performance Evaluation **37**
4.5 Summary **43**

4.1 Introduction

As has been presented in Chapter 3, we leverage IoV-based crowdsensing to overcome many of the barriers faced, including sensing costs and Small-scale coverage, Thus, we intends to study a MCS framework to achieve scalable coverage in real-time for urban and rural environments, and reducing costs and especially diseases within cities.

Particularly, this work studies a novel type of MCS task that aims to collect sensing results from a specified number of participants in the target region within an urban environment.

This chapter introduces our framework, which relies on IoT solution, designed to identify suspected cases of COVID-19, and addressing its rapid outbreak, leveraging IoV networks, to enhance the data collection in large crowds, and ensure full coverage in real time. Our contributions are summarized as follows:

- Designing an IoV-based Architecture for pandemic corwdsensing, empowered by embedded sensors, to monitor and localize the infected persons, and thus avoid or alleviate pandemic outbreak (Section 4.3), fits for urban and rural areas.

- Using Edge server brings real-time data processing and data storage, and using visual display of data , for delivering information efficiently to consumers such as health department directorates, (Section 4.3), and we propose a prevention strategy via smartphone.

- We evaluate the effectiveness of our solution based on simulation results of the QoS parameters in the network, (Section 4.4).

4.2 Related Work

This section presents the most relevant works about how to use IoT in the healthcare system through real-time health monitoring, especially the detection of infectious diseases. Moreover, we highlight some works that use IoVs based approach for remote sensing applications.

4.2.1 IoT for healthcare and epidemics detection

The IoTs have gained significant use in recent years and offered many purposes and benefits for a variety of domains, especially monitoring and healthcare system, which it has worked to improve and increase its efficiency.

Several research works have been carried out to study the benefits of using IoT in health care systems. The authors in [40, 41] reviewed the use of remote sensing via IoT applications in the healthcare field, also discussed addressing some challenging issues for designing smart healthcare systems, including resources and big data management, to increase the efficiency and quality of patient care. Gupta *et al.* in [42], proposed a solution for remote patient monitoring based on wearable body area networks (WBAN) coupled with cloud computing, and deployed a strategy to mitigate inter-WBAN interference, improve energy-efficient and fault-tolerant communications.

However, some IoT solutions couple with other technology such as cloud computing and social sensing are devoted to track and detect epidemics. For instance, the authors in [43, 44] present the use of research engines like google and social media applications to predict and track epidemics of seasonal influenza via social sensing relying on social posts and the numbers of search queries.

4.2.2 IoV-based remote sensing

A connected vehicle equipped with multiple sensors is able to collect a variety of information from the surrounding environment. The majority of research studies have focused on smart cities applications including traffic control and air quality monitoring, etc. For instance, Hu *et al.* in [45], proposed a VSN architecture that estimates the amount of carbon dioxide (CO2) gas emissions remotely, using GSM short messages and vehicle GPS tracker, to map its location data. The edge server is used for both storage and compute. In the same context, the authors in [46], leverage VSN for air quality monitoring in metropolitan areas, implementing a new gathering strategy using rewards for each report received by the edge server, which increases the effectiveness of collecting data under various scenarios. However, the accuracy in the gathering process should be improved, in addition, drivers cannot controlling the sampling rates.

Mathur *et al.* in [47], designed an application named ParkNet, to check the parking availability information using smartphones ultrasound sensing technology combined with GPS trackers and report it to a centralized parking server. The power constraints of ultrasonic sensors are the primary factor that affects the performance of this architecture.

4.3 System Design

Figure4.1 shows the design of our proposed vehicle-based remote sensing platform composed of two sections: a front-end section represented in smart vehicles equipped with GPS and thermal camera sensors, and employing wireless sensor networking technologies to collect data in real-time, and a back-end database on a server at the edge of the network. The goal of this platform is to provide real-time, cost-effective, secure, and large-scale data acquisition, in order to identify the most affected areas with the Covid-19 pandemic, which helps in decision-making by early intervention and thus contain the spread of the virus.

The proposed vehicle-based remote sensing platform consists of three stages as indicated in the following.

- **Sensing and data analysis stage:** the patrol smart vehicle (i.e., police cars, ambulances, etc) has a built-in thermal camera for helping remotely sense and processes some vital signs, including body temperature and breathing rate, by tracking the temperature of the nasal area changes during inhalation and exhalation of the pedestrians on the street. Thermal camera imaging is semi-automatic to allow in some cases the operator (e.g., policemen or ambulance men) to control it and help intervention, in order to reach the narrow corners and to sweep the wide areas.

  The sensed data is subject to a process of analysis, which is intended to identify the infected persons with Coronavirus (COVID-19), by checking if their bodies temperature is higher than 38°C, which mean it exceeds a normal threshold [1] and they have respiratory complications associated with potential COVID-19 (i.e., increased breathing rate [48]).

- **Transferring Data stage:** The data sensed by vehicles can reach successfully its destination, it must be followed two steps. First, vehicles transmit real-time information to the base station, leveraging Vehicle-to-Broadband Cloud (V2B) communication, using 4G/5G LTE technology to cover both urban and rural areas, which in turn, forward it to the edge server through wired communications, as shown in Figure4.1.

[1] The normal body temperature or normothermia ranges from 36.5°C to 37.5°C.

- **Aggregation and decision stage:** the process of data aggregation and analysis collected by remote sensing in real-time is done at the level of the edge server. The output of this stage includes the creation of geographic heat-map reports, which can be used by some stakeholders such as the Health Ministry Dept. for a prompt assessment. This allows making early decisions that can prevent or slow the spread of infections like COVID-19, including putting people in quarantine or under medical surveillance, performing viral tests, and disseminating alert announcements, for example, through the use of online social networks (OSN) or SMS.

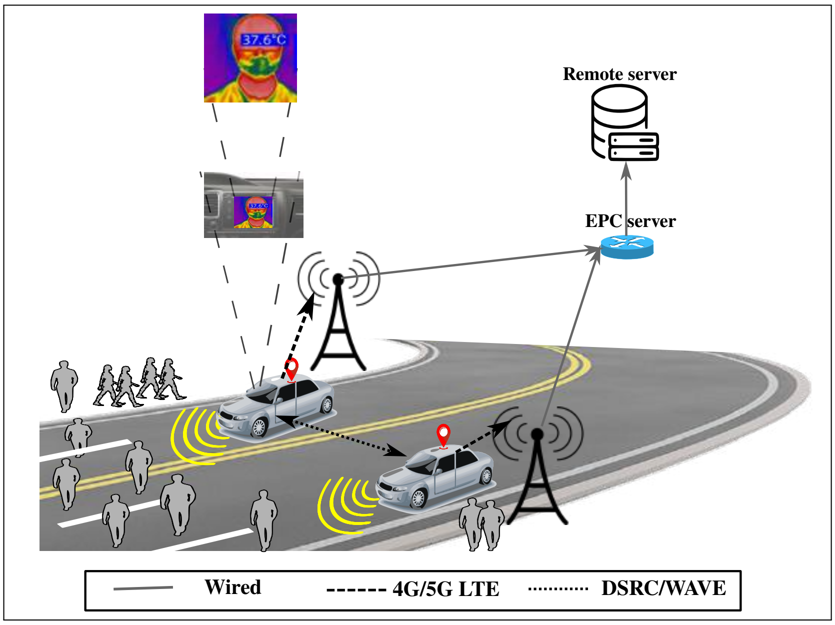


Figure 4.1: The proposed design for identifying suspected cases of the COVID-19 virus.

4.3.1 Prevention Strategies

Figure4.2 shows how to leverage our design to create prevention strategies using OSN/SMS technologies as follows:

The OSNs-based solution. this design integrates social media as an efficient way, to spread the information about the existing or potential risks of infections in specific geographical areas, in the online environment so quickly. As smartphones are ubiquitous in modern societies, with the increasing use of social applications, which makes OSN had a great impact on prevention support. therefore, our framework promptly shares the information collected at the edge server level, using official social media accounts such as those of the Health Ministry, to ensure the reliability and credibility of the information.

The SMS-based solution. The short message service (SMS) alert or text message is the easiest, cheapest, and most widely available technology, to notify people of possible exposure to the infection, so people receive notification from Health Ministry in coordination with all mobile network operators (MNO). Thanks to the Geo-location information provided by smartphones, it will be possible to notify only people within the endemic areas. The notification includes instructions with multiple suggestions, such as maintaining physical distance, hand washing, wearing a face mask, limiting social interaction, etc. The privacy of pedestrian users is preserved, because of the anonymity that is used in the collecting process, and only the location of possible infection cases is notified.

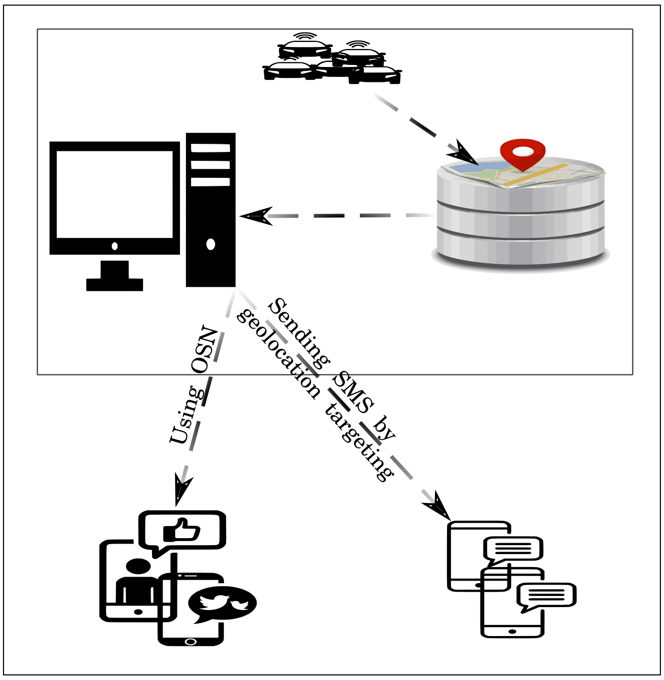


Figure 4.2: Exploiting the proposed architecture for prevention purposes.

4.4 Performance Evaluation

4.4.1 Dataset

The quantitative analysis of the proposed system involves generating a synthetic dataset of suspected cases, depending on the number of cases infected with Covid-19, that are daily reported in a particular geographical region and a correspondent population density. Through this information, we set an infection threshold $\mathcal{T}$, equal to the ratio between the discovered infection cases and the inhabitants in that zone. For example, given a population density of $\mathcal{P}$=1000 persons, among them $\mathcal{C}$=10 infected cases have been discovered, So we specify the threshold as: $\mathcal{T}= \frac{\mathcal{C}}{\mathcal{P}} =$ 0.01. We must take into consideration this threshold when we create the synthetic database to reflect the ratio of infected pedestrians in a given geographic region.

The Algorithm 1 presents the random generation of body temperature and breathing rate values. When a pedestrian body temperature exceeds 38°C, and he had a decreased oxygen saturation or increased respiratory rate (i.e., more then 30 breaths per minute), he is considered as a new potential infected with COVID-19. Assign these values to potential infected pedestrians, by taking into account the population density in each area(i.e., taking Annaba city, Algeria divided in five regions according to the population density as a sample), as indicated in Table 1.

Table 1: Synthetic dataset of suspected cases.

	Pedestrians	**suspected cases**
zone 1	38000	6500
zone 2	24000	1200
zone 3	12000	4500
zone 4	5000	2000
zone 5	4000	700

To improve data-driven decision-making and create an early intervention. The generation of a geographic heatmap to display the distribution of the pandemic across

Algorithm 1 Generation of synthetic dataset

Input : $\mathcal{P}$ a number of pedestrians at a particular location and $\mathcal{C}$ a number of reported cases per $\mathcal{P}$ population.

Output : Number of suspected cases in this location.

Let s be the number of suspected cases initialized with 0,

Let T be the infection threshold, where $\mathcal{T} \leftarrow \frac{\mathcal{C}}{\mathcal{P}}$:

while $\frac{s}{\mathcal{P}} \leq \mathcal{T}$ **do**

    Assign random value to fever symptom

    Assign random value to breathing rate

    if *(the fever generated value > 38 ℃)* ***and***
    (the breathing rate generated value > 30 *rpm)* **then**

        Add as suspected case.

        Increment s by one.

    else

        Discard the new case.

    end

end

the different areas of Annaba city, Algeria, is of crucial importance, considering the degree of its spread represented with gradient color, as depicted in the Figure4.3

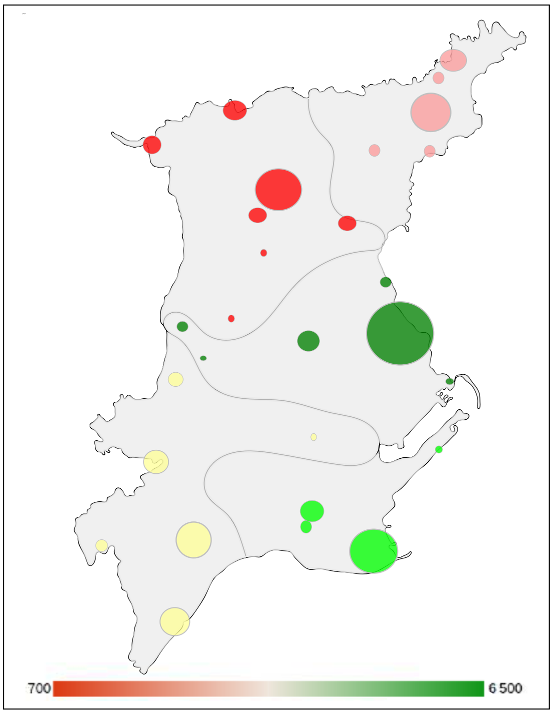


Figure 4.3: Suspected cases of viral diseases heatmap.

4.4.2 Simulation and results

In this subsection, we evaluate the performance of our proposed scheme based on a synthetic dataset as an input of vehicles, using the network simulator-3 (ns-3), and the simulation of urban mobility (SUMO). To send the information of each suspected case to the edge server leveraging LTE technology, we assume that each data packet carries localization information, body temperature, and breathing rate of only one case. The major parameters used in the simulation are given in Table2. The Vehicles' density is ranging from 2 to 40 for each geographical zone, the Nakagami-m distribution is the suitable model used to reflect Non-line-of-sight (NLOS) channel conditions.

Table 2: Parameter Setting in the Simulation.

	Simulation parameters	Values
General	Communication technology	LTE + wired
	Simulation Time	50 seconds
	Vehicle density	2, 6, 10, 20, 40
	Number of eNB	2
	Number of pedestrians	4000, 5000, 12000, 24000, 38000
	Ratio of contaminated pedestrians	17.5%, 40%, 37.5%, 5%, 17.1%
	Data packet size	1024
LTE	Propagation loss model	Nakagami
	LTE data packet type	TCP
	Data rate/RB allocation	DL (50)/UL (50)
	Transmission power	eNB (49 dBm)/UE (23 dBm)

A. QoS metrics:

We evaluate the quality of service (QoS) metrics against a varying vehicular density from 2 to 40 using TCP traffic pattern, which provides reliable stream delivery of data [49], the obtained results of our simulation are presented by measuring different QoS metrics as depicted in Figure4.4, such as end-to-end delay, packet delivery ratio, throughput and delay variation, which can be defined as follows:

- **End-to-end delay(E2E delay)** is the total time required for a packet to reach its destination. the mean E2E delay is computed as the average ratio between the sums of all delays and the total number of received packets, as follows:

$$\mathbf{Mean\_E2E\_delay} = \frac{\sum \mathbf{delaySum}}{\sum \mathbf{receivedPackets}} \tag{4.1}$$

- **Packet delivery ratio (PDR)** is the rate of packets successfully transmitted over the total transmitted. see equation (4.2)

$$\mathbf{PDR} = \frac{\sum \mathbf{receivedPackets}}{\sum \mathbf{transmittedPackets}} \tag{4.2}$$

- **Throughput** represents the amount of data successfully received by the edge server per unit time, measured in kilobits per second (kbps), as shown in equation (4.3).

$$\mathbf{Throughput} = \frac{\sum \mathbf{receivedData_{kbps}}}{\mathbf{t}_{last} - \mathbf{t}_{first}} \tag{4.3}$$

where: $\mathbf{t}_{last}$ represents the time of last packet send, while $\mathbf{t}_{first}$ repreents the time of first one.

- **Delay variation (or delay jitter):** represents the variation in the end-to-end delay of packets, caused by some factors such as congestion. The mean jitter can be given in the formula 4.4.

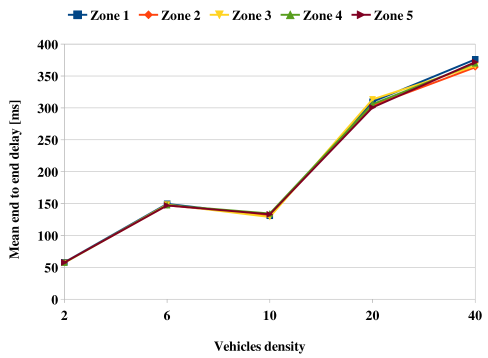
$$\mathbf{MeanJitter} = \frac{\sum \mathbf{jitter}}{\sum \mathbf{receivedPackets - 1}} \tag{4.4}$$

From Figure7.5a we can observed that the delay increases with the number of vehicles, this occurs when a network becomes congested. Despite the variation in the number of vehicles, the packet transmission time is less than 1 second, which means that the transmission occurs in real-time.

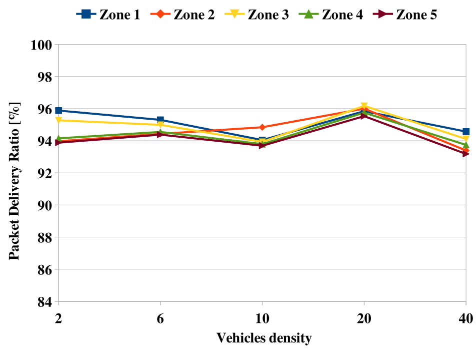
As shown in Figure7.5b, it can be observed that our design is able to reach and maintain a high packet delivery ratio, even when the vehicles density increases.

Figure7.5c shows that, reasonably, the throughput increases with the vehicle density, because of the great number of packets successfully delivered to the edge server.

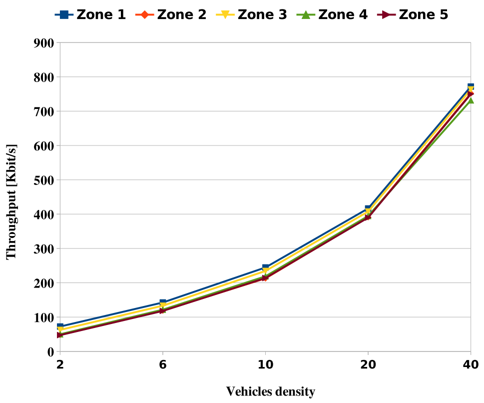
Figure7.5d shows that the delay jitter increases with the vehicles' density, due to the higher packets congestion.



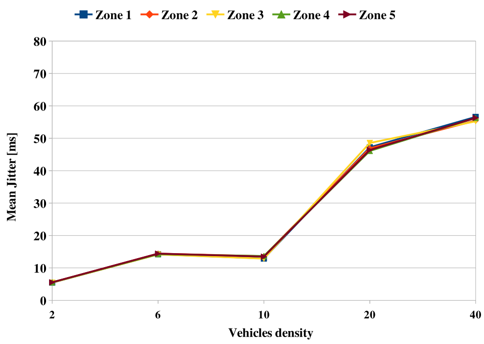
(a) Mean E2E delay variation according to vehicles density



(b) Average PDR variation according to vehicles density



(c) Throughput variation according to vehicles density



(d) Mean delay jitter in respect of vehicles density

Figure 4.4: QoS metrics ((a) E2E delay; (b) PDR; (c) Throughput; (d) Jitter)

B. Accuracy measures:

Infected persons with a very low grade or without symptoms, or who had seasonal flu symptoms can affect the remote sensing accuracy, this can lead, to the occurrence of false-negative or false-positive results.

- False negatives (FN): A test result which falsely indicates the absence of a condition or an outcome.
- False positives (FP): A test result which falsely indicates the presence of a condition or an outcome.

To simulate the accuracy of tests to identify infected people, we consider zone 4 composed of 7000 pedestrians, among then 5000 pedestrians tested negative for COVID-19, and the rest tested positive. We assumed that false-positive of tests are 4% and false-negative of tests are 1%, as mentioned in the existing literature about remote sensing through thermal imagining cameras [50, 51], we get the following results:

$$FN = P(D \cap \overline{P}) = 20, \tag{4.5}$$

$$FP = P(\overline{D} \cap P) = 200, \tag{4.6}$$

Where:

P : the sensing result is positive.
D : the pedestrian has the disease.

The potentially infected people in this design are identified when the body temperature is absolutely greater than 38°C, and the breathing rate is higher than 30 rpm. If pedestrians are tested positives, they require treatment, isolation, and quarantine, thereby, breaking the chain of infection. However, having false positives is preferable to having false negatives, which lead to bad outcomes on public health.

The graph in Figure4.5 displays the range of FP and FN values around a specific body temperature(i.e., 38°C). Similarly, for the breathing rate, we take the threshold as 30 rpm.

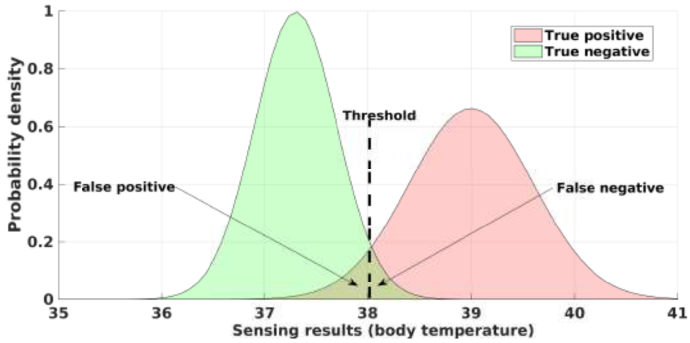


Figure 4.5: Graph to represent false negative and false positive.

4.5 Summary

This chapter presented the use of IoV-based remote sensing framework to identify and control the outbreaks of infectious disease like COVID-19. The chapter started by identifying endemic areas, using sensors devices, including thermal cameras and GPS trackers built into smart vehicles, to save time and a allow large-scale coverage, which were the targets of this thesis Then, continued by presenting a great and cheap way to notify people of possible exposure to the infection, using OSN/SMS based prevention. The experiment results verified the efficiency of design in terms of QoS metrics for mobile crowd-sensing tasks.

"Distance sometimes let you know who's worth keeping and who's worth letting go."

— Lana del Ray

Chapter 5

DeepDist:Deep learning-based physical distancing violations in IoV ecosystem.

Contents

5.1 Introduction 44
5.2 Related Work 45
5.3 Preliminary: An overview of Faster R-CNN algorithm 47
5.4 DeepDist Framework 49
5.5 Evaluation 51
5.6 Discussion 52
5.7 Concluding Remarks 54

5.1 Introduction

Crowd management systems play a key role in today's smart cities for many applications through cameras devices, such as urban planning, disaster warning, emergency evacuation, and especially, building prevention mechanisms for widespread viral diseases such as Coronavirus 2019 (COVID-19), and relying on Internet of Things (IoT) solutions. However, most of these applications need a real-time warning mechanism that allows safe crossing by practicing physical distancing, avoiding high expenditure for installing cameras in most places, in addition to the absence of any pedestrians on some corners makes them useless.

In this chapter, we propose a physical distancing notification strategy, through the use of connected vehicles' build-in cameras switching system to work at night and day, enhanced by a deep learning algorithm using Faster R-CNN, to detect and measure the distance violation between objects of the same class in real-time. In summary, this chapter makes the following contributions:

- Design a new framework to identify accurately the physical and safety distancing violations among pedestrians and inter-vehicles using SIoV, (Section 5.4).
- Getting optimal planning decisions for crowd management using advertising boards and vehicles' dashboards,and Leveraging cloud framework to count the number of violations in each area, and creating a Geo-heatmap to identify and avoid crowded places,(Section 5.4).

5.2 Related Work

Our work consists of practicing physical distancing using an object detection mechanism, which aims to prevent the spread of the epidemic. In this section, we discuss current approaches related to our work.

5.2.1 Measuring Physical Distance

J.K Kalso *et al.* in [52], simulated the effect of four different social distancing intervention strategies to protect against potential influenza pandemic. The results proved that these strategies give greater effectiveness, unless they are applied together for an extended period of time. Despite the importance of these measures, personal freedom remains a challenging problem.

Zhang *et al.* in [53], proposed a method for measuring the distance of traffic obstacles in real-time from different angles to obtain 3D target information, which helps to assist with blind-spot monitoring during the reversing process, using a deep learning model and binocular vision methods. However, in this study, some detection issues related to different types of obstacles as well as target details such as speed, direction, and angle are confronted and must be addressed to increase accuracy.

Zaarane *et al.* in [54], studied a measurement system for autonomous vehicles using a stereo camera to detect and measure inter-vehicle distance. Despite the high accuracy shown by the extensive experiments. However, this proposal suffers from being slow in objects detection, in addition to the issue of matching the same vehicles in both cameras in the multi-object environment.

Huang in [55], presented an effective driver assistance system using a built-in camera, for detecting and locating moving vehicles, and then estimating inter-vehicle distance, leveraging the Histogram of oriented gradients (HOG) feature and Support Vector Machines (SVM) Classifier. The performance evaluation shows that even with limited vehicle detection in high scene complexity, this design can achieve a high detection rate. In the same context, Sasaki *et al.* in [56], put forward an inter-vehicle distance estimation method based on the similarity relationship between the license plate and the captured image, using a smartphone camera. This design needs to minimize the error rate variance by increasing image resolution and sacrificing a little computational time.

5.2.2 Object Detection and Recognition

in the last decade, deep learning has transformed computer vision into numerous solutions and deployed it to different industrial applications, such as object detection, classification, and tracking. Recently studies that focus on DL algorithms and computer vision have become ubiquitous. Wanger *et al.* in [57], design of two models for pedestrian detection using Fast-R-CNN algorithm, the proposed architecture has the capability to support a real-time detection system for visible and thermal streaming video. This design achieves great accuracy, but its need for the GPU to accelerate the calculation speed makes increases the cost of hardware, which makes it not practical.

Punn *et al.* in [58],propose an effective solutions using DL-based pedestrians detection and monitoring social distancing violations, using the fixed cameras in public areas, to mitigate the spread of the COVID-19 pandemic. Although the high accuracy and precision that the YOLO v3 algorithm gives, the design and deployment of such systems require a huge dataset for the training phase.

Aqib *et al.* in [59], discuss an emergency evacuation framework, using DL-based traffic state forecasting and big data analytics to suggest freely flowing routes in smart cities. This model achieves high accuracy but fails when changing the dataset.

5.3 Preliminary: An overview of Faster R-CNN algorithm

5.3.1 Architecture

Convolutional Neural Networks (CNN, or ConvNet) are widely used deep learning architecture, which consists of multiple layers including input Layer, Hidden Layer (convolutional layer, pooling layer), fully connected layer, and Output Layer, to filter input volumes to higher levels of abstraction, and primarily used for computer vision and visual analysis imagery applications. Faster R-CNN is being among the top trending DL algorithm, that relies on CNN architecture, such as You Look Only Once (YOLO) and Single Shot Detector (SSD), for real-time object detection, and offering speed-accuracy trade-offs [60]. The overall structure of faster R-CNN can be considered as the combination of twos parts: The region proposal network (RPN) and the Fast R-CNN as shown in Figure5.1.

- *The RPN as a region proposal algorithm:* it replaces selective research and defines "where" to search to reduce the computational requirements, using anchors as reference boxes for generating proposals at the regression layer level, while the classification layer gives the probability that each anchor shows an object, then remove similar bounding boxes results that match the object class predictions using non-maximum-suppression (NMS), thus generating high-quality region proposals, which are used by Fast R-CNN for detection.

- *The Fast R-CNN as as a detector:* the proposal regions provided by RPN and the input feature maps are passed to the Fast R-CNN algorithm via the region of interest (RoI) pooling layer to reduce the proposal feature maps, then fed into the fully connected layer, which in turn passed to the softmax and regression layer for classification and bounding boxes prediction, respectively.

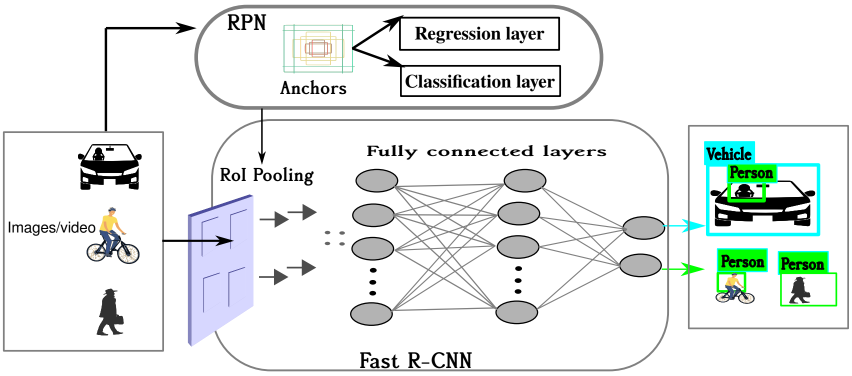


Figure 5.1: The general architecture of faster R-CNN.

5.3.2 Training and Loss Functions

After extracting feature maps from the input image using ConvNet, and passing it to RPN for training by specifying a label to each anchor to form a mini-batch that contains 256 anchors, and generating region proposals. These proposals are fed to the Fast R-CNN module to train, Region of Interests (ROI) pooling is used to bring down all similar proposals. The output is flattened to the fully connected layers in order to classify any predict the bounding boxes. and the process repeats.

For training RPNs, we assign a binary class label (object or background) to each anchor. There are two losses in Faster R-CNN: one is classification (category) loss L_{cls}, and other is regression (bounding boxes location) loss L_{reg}, as shown in the following[60]:

$$L(p_i, t_i) = \frac{1}{N_{cls}} \sum_i L_{cls}(p_i, p_i^*) + \lambda \frac{1}{N_{reg}} \sum_i p_i^* L_{reg}(t_i, t_i^*),$$

where:

i	the anchor index.
λ	a balancing parameter.
p_i	the probability of predicting the object.
p_i^*	the ground-truth label, $p_i^* = \begin{cases} 1 \text{ if object} \\ 0 \text{ if background} \end{cases}$
t_i	the predicted bounding box coordinates vector.
t_i^*	the ground-truth box if (+) anchor.
$N_{cls},\ N_{reg}$	normalized numbers.

5.4 DeepDist Framework

This section elucidates a novel DL-based framework called DeepDist, that aims to determine whether they are following 'physical Distancing' guidelines of maintaining a minimum distance to prevent COVID-19 spreads from an infected person to others. DeepDist combine visible and thermal camera for pedestrian and vehicle detection, The use of thermal cameras is motivated by the fact that they can produce an image with no or few distortions during darkness and/or difficult weather conditions (e.g. fog/rain/snow), whereas the changes in daily maximum or minimum temperatures can affect thermal performance, which requires the inclusion of a visible camera, for its good performance in daylight, with some limitations in in low illumination conditions and at night. However, the disadvantages of both types can be overcome by adopting a switching cameras system.

The proposed framework is distinguished from that of literature that we also involve the highly mobile SIoV, this allows full geographic coverage at the city level rather than a small area. So, relying on the visible and thermal cameras that each car equipped, and using Faster-RCNN algorithm to identify and locate two objects(pedestrians and vehicles), then, measuring the distance between objects of the same class to monitor the physical distance, and generate alert messages in the case of distance violations, as shown in Figure5.2

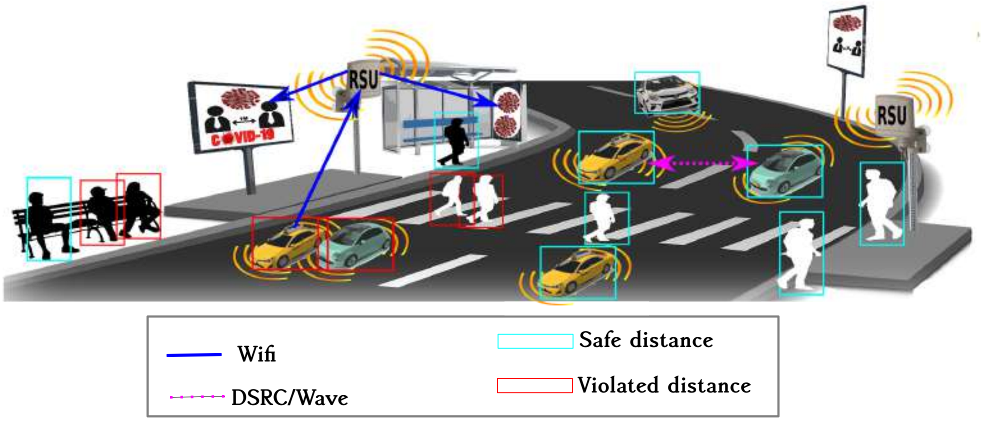


Figure 5.2: The general architecture our proposal approach.

– **Case 01:** vehicles are able to automatically detect a violation of physical distancing among pedestrians in real-time, and send a live alert to the nearby roadside units (RSUs), which forward it the closest digital advertisement boards to display their HD videos/images, which indicate to the need to practice physical distancing to ensure their safety and maintain public health during the pandemic.

– **Case 02:** our scheme also leverages the objects detection of the same class, which allowing cars to be detected and identify those are close to each other, that move in parallel lines, and broadcast an alert, using Decentralized Environmental Notification Message (DENM), which include the position of the concerned vehicles that violated the physical distancing. Figure5.3 illustrates exactly where the safety distance violation alert in the situation container of the DENM.

– **Case 03:** the cloud-based solution in our design provides an opportunity to deploy a good solution to avoid crowded areas, by sending alert messages accompanied with their Geo-location via vehicle-to-broadband cloud (V2B) communication to the central platform, and there the number of times that an alert has fired in a specific area, Then it is displayed using a heat map, which allows the user to build up prior knowledge about a condition a road, area, bus stops if they are crowded or not.

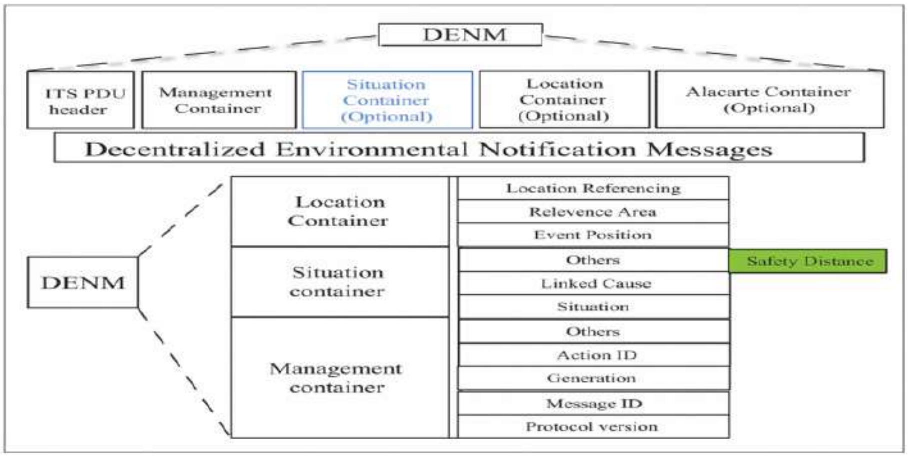


Figure 5.3: Extended DENM to support safety distance violation alerts.

5.5 Evaluation

In order to validate the effectiveness and feasibility of the proposed system, performance evaluations are conducted using two open-source frameworks, Network Simulator (NS-3), and the Simulation of Urban Mobility (SUMO) to generate realistic mobility behavior for vehicles.

IEEE 802.11p protocol is used for wireless communication in vehicle-to-vehicle and vehicle-to-Infrastructure communications, which is considered as the updated version of IEEE 802.11b standard that works on the data link and physical layers and enables the communication between high-speed vehicles. We assume that each vehicle sends an alert packet to the nearby RSUs, which in turn forward it to the close digital advertising boards to display HD alert images from its stock, once a physical distancing violation is detected. In this Section, we measure in the first part different quality of service (QoS) such as end-to-end delay, packet delivery, with various numbers of vehicles, ranging from 2 to 100 using TCP traffic pattern and two rays ground as a propagation model. The precision and recall scores are calculated in the second part, which is used to measure the accuracy of information retrieval, classification, and identification within a computer program.

Besides the QoS-related metrics, precision and recall are the two key metrics to highlight the sensitivity state of our detector and evaluate model accuracy. Precision

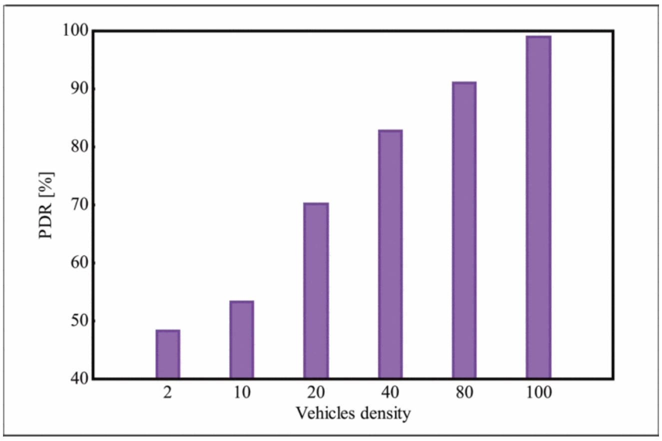


Figure 5.4: Average PDR variation according to vehicles density.

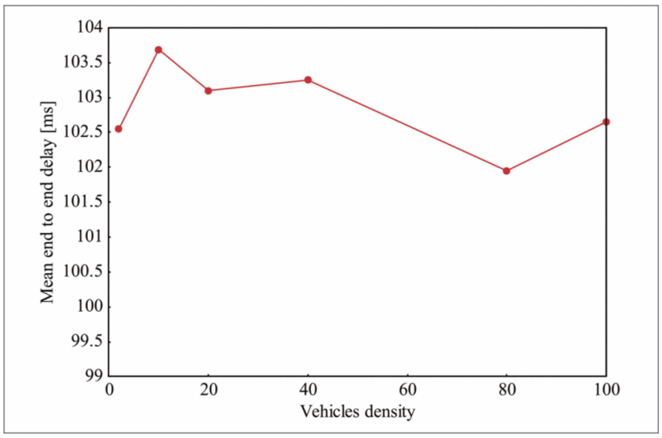


Figure 5.5: Mean E2E delay variation according to vehicles density.

is defined as the accuracy of the classifier to find all the instances of the positive class, while recall detects the largest possible number of positive cases.

A large number of existing studies have examined object detection in different classes using Faster R-CNN and calculated precision/recall (PR) to show up the sensitivity of the detector. The Precision / Recall (PR) curve using Stanford Vehicles' Dataset; is shown in Figure 3, and the mean average precision(mAP) value is obtained from faster R-CNN, which approximately reaches 0.76.

5.6 Discussion

This research work proposes a novel application of SIoV data and capabilities at the service of road users and the local community, by monitoring the physical

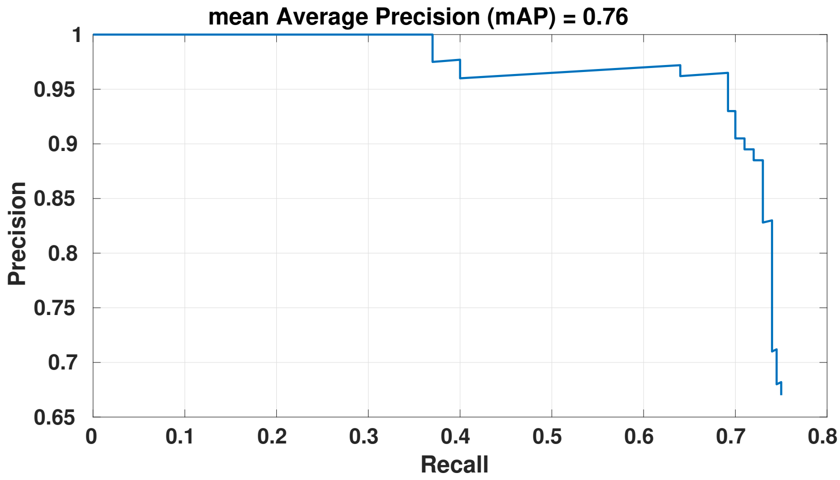


Figure 5.6: Precision/recall (PR) curve for Faster R-CNN.

distancing for crowd and traffic management and alerting in cases of violations in real-time, with great accuracy, we will discuss some issues that may be encountered, and point out some future potential research directions.

pedestrian density: The deep cameras can smoothly track moving objects, measure physical distancing easily in a low and medium density scene with great accuracy; meanwhile, the overcrowding problems have become a common issue in the urban areas, especially in big cities, which affects the visibility of cameras, where some of the pedestrians become hidden together, and become invisible even for the human observer, therefore, it seems very difficult to put bounding boxes to all in this dense scene. As an alternative solution, we should think about using drones in these crowded places to get a full view and overcome the overcrowding problems.

Traffic density: indeed, all modern cities suffer from traffic congestion in all streets, especially at peak periods, which makes physical distancing between vehicles a challenging task as they are subject to driving regulations, in comparison to pedestrians that can move freely. Therefore, it is necessary to think of an alternative solution to prevent the spread of the virus, such as focusing on monitoring if the car windows are open at the time of overcrowding and sending alerts to close them.

Processing costs: image processing and annotation are expensive in both CPU/GPU and storage. The use of buses (with a pre-defined path) and taxi/user (with random path but popular) can help in increasing the benefits of edge computation by providing an extra server to enhance the computation capabilities of surrounding cameras. The compute-less paradigm can be leveraged as a solution, to enhance the computation deadline and computing resources, An edge server can store not only the content but also the computation input/output.

5.7 Concluding Remarks

In this chapter, we design a deep learning-based framework called DeepDist, that leverages the SIoV paradigm to detect and avoid physical distancing violations at a large scale, which used Faster-RCNN to track both types of objects(pedestrians and vehicles) through a switching cameras system. In this architecture, digital advertisement boards are applied to display any alert about violation via V2I communication, or directly via vehicle-to-vehicle communication. Our DeepDist framework was evaluated via Qos and precision/recall metrics. The experiment results verified the efficiency and effectiveness of DeepDist to track objects and avoid physical distancing violations at a city scale.

"The better the coverage, the more discriminating the viewer."

— JESSICA SAVITCH

Chapter 6

GUAVA: Ground vehicles assisted UAV crowdsensing frAmework to combat the spread of covid-19.

Contents

6.1 Introduction **55**
6.2 Existing Work **57**
6.3 The GUAVA Framework **59**
6.4 Evaluation **68**
6.5 Discussion **73**
6.6 Concluding Remarks **74**

6.1 Introduction

We introduced in the previous chapters(4 & 5) two schemes for mobile crowdsensing, that use ground vehicles equipped with intelligent sensors technology, to fight infectious diseases, like Covid-19, and/or predict their spread. The introduced vehicle-based crowdsensing solutions, unlike other mobile devices, do not have strict energy, storage, and processing constraints. Besides, they have the capability to remote sensing at the city scale in real-time. which proved to be highly efficient in terms of scalable coverage. however, ground vehicles' movements depend on the roads' topology that restrict their crowdsensing capability, especially in hard-to-reach areas, like stadiums, public parks, etc. which may not be feasible to ensure full coverage.

In this chapter, we suggest adding Unmanned Aerial Vehicles (UAVs) to overcome the aforementioned issues, which can leverage high mobility and minimal costs to collect data at a large-scale, and work elastically to cover everywhere and everything.

The prevalence of UAVs equipped with thermal cameras and GPS devices and other smart sensors, and their extensive use in various applications, especially in order to stop or reduce Covid-19 pandemic outbreak, helped to maintain physical distance and sensing the vital signs of pedestrians, by exploiting its high dynamism and great flexibility for collecting data, then analyze it on Cloud level, to make better decisions. However, the low flight time of UAVs, especially with the smart payload, brings problems to occur in sensing tasks. In order to overcome this problem, the current study or GUAVA i.e., a ground vehicle (GV) assisted UAV crowdsensing framework, that performs sensing tasks and real-time data collection across large-scale areas to prevent the spread of infectious diseases. with the possibility that GV delivers fast wireless charging to UAV, as illustrated in Figure 6.1. Specifically, our main contributions are:

- We present GUAVA (Ground vehicle assisted UAV crowdsensing frAmework), which is capable of reliably sensing pedestrians' vital signs (body temperature and breathing rate) under different conditions using a thermal camera, which allows discovering the presence of infectious diseases like Covid-19 and ensuring full geo-coverage in smart cities, and even in rural regions, hard to reach areas(e.g., muddy roads, narrow streets of old cities, parks and stadium, etc.), (Section 6.3).

- The thermal camera's facial recognition technology is used to avoid duplicated data collection, and is used for other identification applications (such as finding missing children); moreover, to deal with the energy constraint, we assume that GVs carry wireless power chargers to supply UAVs, (Section 6.3).

- Through extensive simulations, we evaluate the effectiveness of our GUAVA solution for collecting data in real-time and in a realistic setting and comparing it against benchmark schemes, (Section 6.4).

6.2 Existing Work

6.2.1 Vehicle-based crowdsensing

In recent years, the rise of solutions using vehicular crowdsensing applications (e.g., traffic monitoring, healthcare services, etc) is attracting huge interest because of the large amount of sensed data taking advantage of its high movement and dynamism.

Road traffic apps. Wang *et al.* in [61], present a vehicle-based MCS system consists of participants(vehicles) and a cloud management medium as a recruiter to distribute the sensing tasks and predict the trajectory of public transport. The vehicle that participates in crowdsensing receives rewards based on its contribution to the sensing tasks. Although, this design is cost-effective, it suffers from the problem of selecting participants in a large city, in addition to reliability issues, because of some conditions such as drivers' skills, road status, etc.

TPSense is proposed by Xu *et al.* in [62], which is a trustworthy vehicular crowdsensing framework, that leverages roadside units (RSUs) acting as hot-spots and fog computing for processing and storing event reports, and sharing/receiving information with nearby vehicles and with remote cloud server. Privacy is preserved in this design via using blind signature technology. In the same context, Shao *et al.* in [63], propose a vehicular crowdsensing approach for traffic monitoring, by leveraging road topology, where the participant vehicles in the sensing tasks offload their data at road junctions (sponsor vehicles), which apply local processing after forwarding it to the central server to update the traffic vision.

Healthcare applications. the authors in [64], discuss the use of vehicular applications in the public transportation purpose, to collect health information of the passengers and their travel history, By identifying infected people who did not respect the quarantine procedures upon boarding and informing the health authorities, which help fighting pandemics outbreaks.In the same context, The authors in [37], leverage vehicular sensing power, and discuss the use of thermal cameras with GPS devices embedded on the ground vehicles to identify infected persons based on their two vital signs (body temperature, breathing rate), and localize them, the sensed data is collected in an edge-based database of COVID-19 pandemic, hence, creating a heat-map of infected areas to take prior measures. while Sahraoui *et al.* in [38], use build-in vehicle sensors and deep learning technologies to detect physical distancing violation in real-time, among pedestrians and parallel vehicles in the same line, then rely on the digital advertising panels to notify about violation.

6.2.2 UAV-based crowdsensing

Monitoring in samart cities. Elbarka *et al.* in [65], proposes a distributed trust-aware MCS technique based on UAVs for real-time urban traffic monitoring, The system works in coordination with GVs and RSUs, and receives Awareness Messages, including the current position, speed and direction of the vehicles as input, then provides crowd-related data without additional overhead. One of the most recent works proposed in [66], called PERCEIVE inspired by recent software-defined networking (SDN) technology, where a Swarm of Small UAVs is deployed for continuous video surveillance of crowds in urban areas, and offloading data on the edge, while the swarm plan is controlled from the cloud. car-mounted mobile charging stations are used, which allow the UAVs to be powered periodically based on an intelligent charging scheduling mechanism. Motlagh *et al.* in [67], propose a face recognition approach in smart cities, using sensors onboard UAVs, The video analytics is performed either on UAVs or offloading the data to a MEC node, which showed high performance in terms of saving energy consumption of UAVs and processing time of face recognition.

Tracking in wild areas. UAVs can do monitoring tasks in wild and hard-to-reach locations very efficiently and rapidly, and gather useful information. In this context, Zhang and Li in [68] present a framework to sense wild regions using UAVs in the absence of network connectivity, leveraging 5G and Long-Range (LoRa) technologies, which allows for collecting data with high performance.

Conversely, the authors in [69] propose using a single UAV, to track wildlife and monitor animals, a cluster of wireless sensor networks (WSNs) is adopted to collect data easily, by introducing two collection methods. The first way depends on the UAV visits all the cluster heads that received the sensing information from their cluster members. The second way, the UAV visits only the sink node that aggregates the data of the cluster heads, which reduces hover time, and thus limits the energy consumption.

Finally, the authors in [70] propose a framework for the deployment of the UAVs in disaster scenarios, through Cloud Assistance. During their hover time, UAVs are configured to record videos of the disaster area and to perform data pre-processing, filtering out unnecessary frames. Then, forward it to the cloud for post-processing.

Monitoring and prevention of epidemics. UAVs can help mitigate logistical problems and provide different healthcare services, using a payload of sensors such as thermal cameras, that can collect data and prevent the spreading of infectious epidemics. The authors in [71] attempt to provide a comprehensive UAV-based networked system to limit the spread of infectious epidemics like Covid-19. This proposed model focus on dividing the monitored geographic area into small zones, and Assigning each singe UAV to each one of them using thermal camera sensor, to check the social distancing and monitor the violation cases, in addition to collecting people vital parameters using their wearable sensors. A smart mobility algorithm is used to improve UAVs movements and avoid collisions. However, the power consumption issue has not been addressed.

6.3 The GUAVA Framework

In this section, we describe in detail GUAVA framework, where the cooperation between GVs and UAVs for sensing and offloading data is presented in the section 6.3.1, leveraging the algorithm of detection and alerting infected people shown in section 6.3.2, where the face recognition pattern is presented in the section 6.3.3.

6.3.1 System model

Figure 6.1 shows the data collection process(measuring the vital signs with pedestrians' location, and their face images) of our proposed GV-assisted UAVs design, by integrating GPS devices and thermal cameras with the Ground Vehicles and UAVs control system, which enables to overcome difficult conditions during the sensing process, such as fog and difficult weather conditions, as well as during the night. GUAVA employs wired and wireless technologies (wireless sensor network, 4G/5G LTE), edge computing, this platform supports U2V, U2I, and V2I communication mechanisms and is able to collect data and upload it to Collection Center for further processing. The goal of this platform is to provide real-time data collection about infected areas for early intervention by stakeholders(health dept., prime ministry, etc).

The main components of the GUAVA model are: entity set(Collector, Collection Center) and relationship set (communication), as described in the following:

Collectors. mobile nodes or collectors(UAVs and GVs) are dispersed in the designated areas, to perform sensing tasks in a cooperative manner. In fact, the GV gathers/senses data from some predefined streets by government and local authorities. Due to the limited capability to ensure full coverage, GVs need to cooperate with UAVs, which are remotely controlled [1] to fly at low altitudes, and deployed to different areas of interest to sense data including rough areas, especially those that are difficult to access by GVs, such as stadiums, parks, etc.

The built-in thermal camera function is not only to capture vital signs. but also, can function as a thermal faces recognition system. These cameras are configured to perform real-time sensing tasks for pedestrians' vital signs monitoring, namely (i) body temperature and (ii) breathing rate, which are widely used to evaluate the degree of infection of respiratory diseases like Covid-19. The breathing rate can be measured by counting the number of the temperature in the nasal area changes during inhalation and exhalation [72, 73]. As a result, the temperature of the

[1] We assume that a remote Control Center is in charge of remotely monitoring the missions of UAVs.

nasal area rises during exhalation and declines during inhalation, and the thermal sensor can capture such a variation. The respiration rate is then determined from the breath-to-breath intervals.

When body temperature reaches more than the threshold (38°C), and increases their respiratory frequency to more than 30 breaths/min, which means that the collectors detect a potentially infected individual. The duplicated sensing data can be avoided by adopting an AI-based face recognition algorithm that uses facial landmarks extracted automatically, such as distance between the eyes, ears shape, and nose size, lip shape, which allows detecting matching faces with high accuracy [74], as will be discussed in the section 6.3.3. The Collectors gather the vital sign data of each person associated with their faces, which allow us to exclude persons with identical faces, and therefore not send their data because they are duplicates.

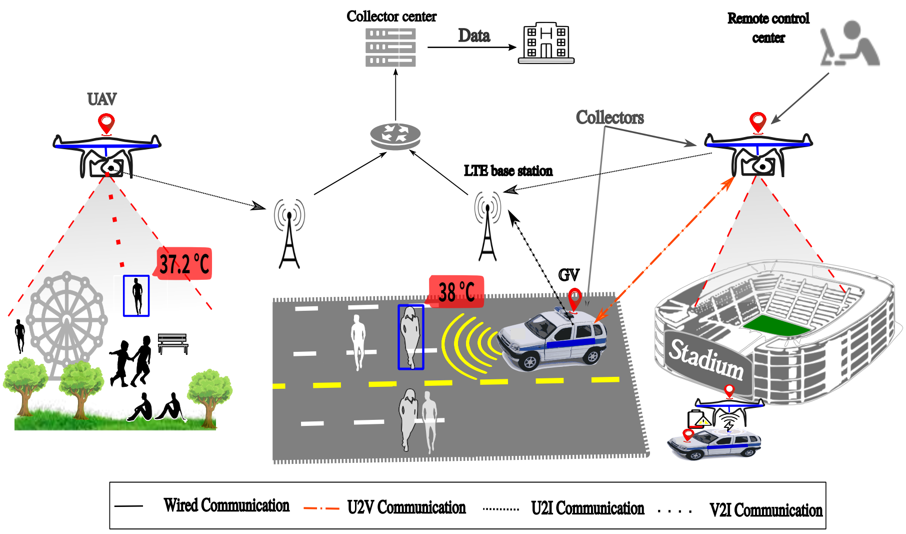


Figure 6.1: General design of our GUAVA framework.

In addition, due to the high battery constraints of UAVs, due to the having sensing payload (camera, GPS devices), speed, environmental constraints, hovering time, batteries type, the use of solar power as an energy resource [75], etc. Therefore, the estimated battery time available for UAVs is not fixed and changes according to these conditions. So a wireless power charger mounted on the roof of every GV is mainly used when the battery level of a UAV drops below a certain threshold, which enables UAVs batteries to be charged without human intervention, just receiving a notification from the Control Center to be redirected to the nearest power supply GV. However, recent studies like [76, 77] have shown that, on average, recent UAVs can fly with carrying sensors payload(eg., camera, GPS device) for up to 30 minutes in one charge, which can be enough time to perform sensing tasks in a specific area, like the full coverage of many hard-to-reach areas. Meanwhile, in large and crowded places, we need to fly multiple UAVs to capture all the data we need. to avoid the interruption of the service and the consequent QoS degradation. To provide complementary coverage and avoid crashes, each UAV is deployed for automatically monitoring a specific area, which does not interfere with other UAVs areas, by following an increasing rectangular helical path in its mobility. For instance, the Figure 6.2 shows that UAVs perform sensing tasks without overlapping, with an average speed of 20 ms for covering $1Km^2$ on a half-hour trip, using the range of 200 m(i.e., rectangle enlargement of 200 m at each step).

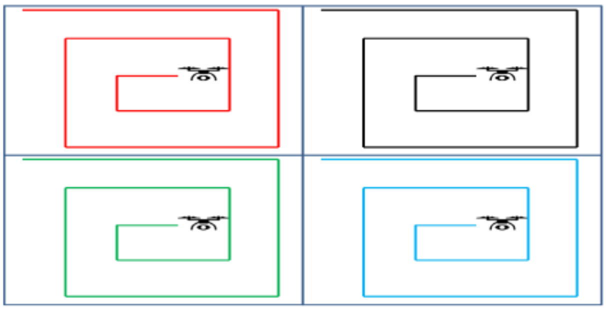


Figure 6.2: Example about UAVs monitoring trajectory.

Collection Center(CC). relying on deploying an edge server for transmitting all sensed data, storing it with low delay processing. Besides GVs sending data, consisting of faces images, vital sign, and geographic information to the CC, the UAV transmit only the body parameters to CC, due to its smaller size i.e., few Kbytes, while the images are offloaded to the GVs during the re-charging process, which in turn forward it to the CC, this makes save UAV energy consumption. A flag is associated with every data unit showing if the sensing source is a UAV or a GV, as well as facilitating the linking of information to each other at the CC level.

The key goal of the CC is to create an overall dataset of the sensed areas, then generates a heat-map for visual showing the spreading of the disease, which allows geographical tracking in time and early intervention by the relevant authorities.

The 3GPP-V2X (vehicle-to-everything) provides enhancements specifically for vehicular communications [78], which allow CC to be implemented as a V2X Application Server according to the Multi-Access Edge Computing (MEC) paradigm, which is characterized by ultra-low latency and high bandwidth as well as real-time access.

Communication. the communication applied in our framework between Collectors and Collection Center is implemented by means of 4G/5G Long Term Evolution (LTE) telecommunication network, which allows the connections between various entities and the CC in real time. As depicted in Fig. 1, flying and ground vehicles transfer the sensed data to LTE base stations, using UAV-to-Infrastructure (U2I) and Vehicle-to-Infrastructure (V2I) communications pattern. which in turn forward it to the edge server through wired links. Beside that, the UAV-to-Vehicle (U2V) connectivity is leveraged to offload the surveillance images from the UAV to the GV. The data in the CC is Labelled based on spatio-temporal characteristics, which allows to filter out the redundant ones.

6.3.2 Remote sensing process

In this section 6.3.2, we propose pseudo-code for the UAVs crowdsensing-based GUAVA model as presented in Algorithm 2, for the formation of GVs/UAVs collaboration network. The Collectors in GUAVA model are represented as a set of GVs/ *i.e.* $G = \{g_1, g_2, \ldots, g_N\}$ with $N \in \mathbb{N}$, and a set of UAVs, *i.e.* $U = \{u_1, u_2, \ldots, u_M\}$ with $M \in \mathbb{N}$. each entity is tasked with sensing a specific area for the purpose of ensuring full coverage and collecting surveillance images and vital signs of the available people, represented as a set $P = \{p_1, p_2, \ldots, p_K\}$ with $K \in \mathbb{N}$. The localisation of the i-th GV (*i.e.*, g_i with $i \leq N$) can be captured by the local GPS device as $\overrightarrow{l}_{g_i} = (x_{g_i}, y_{g_i})$. Similarly, the localization of the j-th UAV (*i.e.*, u_j with $j \leq M$) can be expressed as $\overrightarrow{l}_{u_j} = (x_{u_j}, y_{u_j}, z_{u_j})$.

The k-th person ($i.e.$, p_k with $k \leq K$) is identified by its face, and has a 2D location information compared to GVs $i.e.$, $\overrightarrow{l}_{p_k} = (x_k, y_k)$, and a 3D location information compared to UAVs $i.e.$, $\overrightarrow{l}_{p_k} = (x_k, y_k, z_k)$. In order to be detected, the k-th person must be within the sensing range of the i-th GV $i.e.$, $r_s^{(i)}$, or within the sensing range of the j-th UAV $i.e.$, $r_s^{(j)}$, as follows:

$$\|\overrightarrow{l_{g_i}} - \overrightarrow{p_k}\| \pm \varepsilon \leq r_s^{(i)} \quad \text{OR} \quad \|\overrightarrow{l_{u_j}} - \overrightarrow{p_k}\| \pm \varepsilon \leq r_s^{(j)}. \tag{6.1}$$

where we included the absolute error ε defined as $\varepsilon = \|d_m - \tilde{d}\|$, with d_m [m] as the measured distance and $\tilde{d}$ [m] as the estimated one.

The studies [79, 80] demonstrated that the Long-Wave Infrared (LWIR) thermal cameras on-board of UAVs and GVs can recognize individuals' identity from a distance of 30 m, with high accuracy up to 100%, and decreases as the distance increases, therefore, we set the sensing range $r_s^{(i)} = r_s^{(j)} = 30$ m. for better quality coverage.

As defined in [81, 82], the following threshold values for, respectively, body temperature and breathing rate are used to detect a potentially infectious individual, namely $\tau_1 = 38$ ℃, and $\tau_2 = 30 times/min$.

An individual p_k is considered potentially infected if, and only if, its sensed body temperature $\overrightarrow{V}_{s1,k}$ and the sensed breathing rate $\overrightarrow{V}_{s2,k}$ are higher than given thresholds τ_1 and τ_2, respectively, $i.e.$,

$$\overrightarrow{V}_{s1,k} \geq \tau_1 \text{ AND } \overrightarrow{V}_{s2,k} \geq \tau_2, \tag{6.2}$$

and we also considered the temperature measurement response time $\overrightarrow{T}_{s1}$ and the breathing rate measurement response time T_{s2} should be lower than given thresholds ($i.e.$, $val_{1,2}$ [ms]), such as:

$$\overrightarrow{T}_{s1} \leq val_1 \text{ AND } \overrightarrow{T}_{s2} \leq val_2. \tag{6.3}$$

Algorithm 2 Pseudo-code for the UAVs crowdsensing-based GUAVA model.

Input :
$G = \{g_1, g_2, \ldots, g_i, \ldots, g_N\}$ with $i \leq N$ and $N \in \mathbb{N}$ {Set of GVs}
$U = \{u_1, u_2, \ldots, u_j, \ldots, u_M\}$ with $j \leq M$ and $M \in \mathbb{N}$ {Set of UAVs}
$P = \{p_1, p_2, \ldots, p_k, \ldots, p_K\}$ with $k \leq K$ and $K \in \mathbb{N}$ {Set of people}
$\mathcal{I}_s$ {Set of stored images}
$\mathcal{E}_{u_j}$ {Battery level of UAV u_j}
χ {Warning threshold for batter level}
Output : $\mathcal{I}_s$ {Updated set of suspected cases}
if $\mathcal{E}_{u_j} > \chi$ **then**
  while $\|\overrightarrow{l_{u_j}} - \overrightarrow{l_{p_k}}\| \leq r_s^{(j)}$ **do**
    $\overrightarrow{V}_{s1,k}$, $\overrightarrow{V}_{s2,k}$ { Sensing vital signs from the thermal video}
    Γ {Extract faces from thermal video}
    foreach $\Gamma \notin \mathcal{I}_s$ **do**
      $\mathcal{I}_s \leftarrow \mathcal{I}_s \cup \Gamma$
      if $\overrightarrow{V}_{s1,k} \geq \tau_1$ AND $\overrightarrow{V}_{s2,k} \geq \tau_2$ **then**
        Send the information to the edge server {Possible p_k infected}
      else
        Discard this case
      end
    end
  end
else
  $\min_i d_{u_j,g_i}$, $\forall g_i \in G$ {Compute the distance to the closest GV}
  Move to position $\overrightarrow{l_{g_i}}$ {Wireless charging from the closest GV}
  Offloading $\mathcal{I}_s$ to g_i
end
while $\|\overrightarrow{l_{g_i}} - \overrightarrow{l_{p_k}}\| \leq r_s^{(i)}$ **do**
  $\overrightarrow{V}_{s1,k}$, $\overrightarrow{V}_{s2,k}$ { Sensing vital signs from the thermal video}
  Γ {Extract faces from thermal video}
  foreach $\Gamma \notin \mathcal{I}_s$ **do**
    $\mathcal{I}_s \leftarrow \mathcal{I}_s \cup \Gamma$
    if $\overrightarrow{V}_{s1,k} \geq \tau_1$ AND $\overrightarrow{V}_{s2,k} \geq \tau_2$ **then**
      Send the information to the edge server {Possible p_k infected}
    else
      Discard this case
    end
  end
end

The process of consolidating all the data captured by various collectors as a form of triplet i.e., (infected case, GPS coordinates, face image) to make a single and effective data center free of redundancy to aid early decision making. if there are no strong correlations in the face matching process, the sensed case will be considered as a new potential infectious one. Therefore, GVs send the sensed (and not redundant) vital signs aforementioned and expressed in Eq. (6.2), to the Collection Center, attached with the images of the monitored area. Being represented with the tuple Geo-localization, temperature, breathing rate, which allows us to identify the position of possibly infected people and from it, the endemic areas, to take early countermeasures. On the other hand, if the face matches the other previously felt, the new state is ignored. Only the sensed parameters will be transmitted in real-time by UAVs, and temporarily cache the surveillance images, to save their energy. As depicted in Algorithm 2, these latter will be offloaded during the recharging process to the nearby GV, which in turn forward it to the Collection Center. Notice that the sensing process of the j-th UAV occurs if its energy level (*i.e.*, $\mathcal{E}_{u_j}$) is enough to accomplish this task i.e.,

$$\mathcal{E}_{u_j} > \chi, \tag{6.4}$$

where χ is a given energy threshold. If Eq. (6.4) does not hold, the j-th UAV computes the distance to the closest GV *i.e.*,

$$\min_{j} d_{u_j,g_i}, \ \forall g_i \in G, \tag{6.5}$$

in order to move to that position (*i.e.*, $\overrightarrow{l_{g_i}}$) and recharge its battery level, as well as offload the set of collected images. Notice that the UAV can move autonomously to the nearest GV or also driven by the Control Center.

Finally, when the monitoring process in the designed area has been completed (*i.e.*, the whole area has been covered) the collectors can be assigned to a new one.

6.3.3 Face recognition process

Artificial intelligence (AI)AI is an approach used in our GUAVA framework to perform face recognition, leverages the existing studies in [83], Which depends on the following steps: (*i*) human face detection and features localization, (*ii*) face normalization and feature extraction from it and, finally, (*iii*)– face recognition/classification to find the matching faces based on an existing database. The deep neural network architecture for face recognition consists of a convolutional layer as a set of filters that are trained to extract the features, then the input passes through the pooling layer, for down-sampling its spatial size by extracting important ones [83], while the machine learning architecture for face recognition need a software engineer to identify features. The output of convolutional/pooling layers is flattened and fed into a fully connected layer to classify the images through an activation function.

The use of this architecture offers a significant enhancement and a high performance, bypassing the human level capacity in terms of accuracy. Nevertheless, some issues have been raised affecting the performance of face recognition, including head poses, illuminations, facial expressions, and occlusions [84].

GaussianFace model. to eliminate the transmission of duplicate or redundant sensed information, in this design we propose to use the GaussianFace model, originally described in [85], which offers a facial recognition technology to identify matching faces obtained by onboard cameras with high accuracy. Where the accuracy of the algorithm exceeded 98.52%, which means that it surpassed the human ability (accuracy ratio reaches 97.53%) when using the Labeled Faces in the Wild (LFW) dataset [86]. So After detecting the human face present in a thermal video frame, then the bio-metric facial features are extracted including both mouth corners, eyes, and nose. all images need to be resized to a fixed size i.e., 150×120 pixels depending on the bio-features, and divide each face images into overlapping blocks of 25×25 pixels, the multi-scale uniform LBP histograms are extracted in each block, and create a feature vector, as shown in Figure 6.3 , before the classification stage, this model used as an output function, either Binary Classifier (GaussianFace-BC) or Feature Extractor (GaussianFace-FE) for face recognition, compare them with those in the database and detect and ignore matching ones.

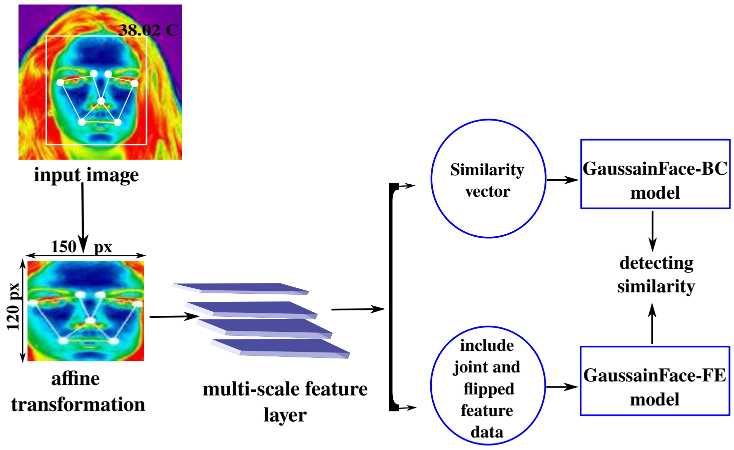


Figure 6.3: GaussianFace architecture to identify matching faces in GUAVA framework.

6.4 Evaluation

In order to validate the effectiveness and feasibility of our GUAVA design, performance evaluations are conducted based on the face recognition process and the Qos of the overall crowdsensing framework.

A bench-marking framework namely BUCST [65] is used, for comparing different metrics including the amount of monitored area over time, traffic overhead. then, we provide optimization and flexible solution focusing on energy saving, by minimizing the number of GVs and UAVs, which requires further investigation.

6.4.1 Simulation setup

In this section, we build a modular scenario of our design as in Figure 6.1, to evaluate the QoS metrics, we use Network Simulator 3 (ns3), where the Collectors sensed data and forward it to the Edge server for more extensive analysis. In particular, we considered a set of pedestrian users walking the streets of the Annaba city center (Algeria). There, GVs are modeled with realistic vehicle behavior through the Simulation of Urban MObility (SUMO) model [87], while, the Gauss–Markov mobility model in a 3D environment [88] is applied to UAV movements, which gives a more realistic behavior for UAVs. Of course, we are conscious of some flight conditions that affect UAV mobility, this allows the control

Table 1: Simulation parameters.

	Simulation parameters	Values
General	Communication technology	LTE + wired
	Simulation Time	50 seconds
	UAVs density	2, 5, 10, 20, 40 [UAV/km^2]
	Number of eNB	1
	UAVs' mobility model	Gauss Markov 3D
	Simulation flight area	200×200 m
	Speed of UAVs (velocity)	20 m/s
	UAVs flight height	30 m
	Data packet size	1024
LTE	GVs propagation loss model	Nakagami
	UAVs propagation loss model	Friis
	LTE data packet type	TCP
	Transmission power	eNB (49 dBm)/UE (23 dBm)
	GVs density	2, 5, 10, 20, 40 [UAV/m^2]
	Simulation ground area	1471.47×1989.8 [m^2]

center to intervene in some cases that require it, such as redirection of the UAV to the nearby GV for wireless charging.

We created a synthetic data set, as in [37], that contains a collection of values of the monitored vital signs for pedestrians likely to have COVID-19, then, we imported this data in ns-3. The configurations of key parameters are listed in Table 1.

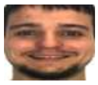
In our GUAVA framework, the evaluation of its effectiveness is based on the performance metrics including throughput, E2E, Packet Delivery Ratio (PDR), and delay variation.

Face Recognition is one of the biometric methods that has recently gained a lot of popularity, so For face matching evaluation, we used the FERET dataset [89] that focus on human faces, were normalized to a size of 150×130 pixels, which helps put all images on equal footing, allows processing, and features extraction. The training dataset for the recognition test consisted of 100 images and we used a gallery as a reference set consisting of 1000 images, and one probe set containing 623 images with different facial expressions, such as smile, astonishment, contempt, fear, and anger.

The Recognition Rate or success rate metric is the total number of correctly identified probe images divided by the total number of probe images. which will be used to estimate the accuracy of the faces recognition process in our framework.

6.4.2 Results

Table 2: Error rate result.

Probe dataset size	Probe images	Error rate
623 images	  Facial expressions	3.69%

To track the effectiveness of our framework, we perform an assessment approach for face recognition is based on the GaussianFace-FE model, to see how accurate is our GUAVA framework, and then investigate its QoS metrics. As depicted in Table 1, we consider the selected dataset and compare it against gallery images. The dataset contains frontal face images with different facial expressions, the face recognition model has an error rate of just 3.69%, which means the GUAVA provides a high success rate with low training. However, the accuracy can be affected under A-PIE variations (Ageing, Pose, Illumination, Expression).

Although the accuracy-efficiency characteristic that the GaussianFace algorithm offers, this approach is slightly delayed (*i.e.*, about 1.02 s per face recognition with i-5 4300U CPU @1.90 GHz processor speed and 4 GB RAM, without dedicated graphic card.). It should be noted that the main idea is face recognition when we look at the overall E2E delay per each crowdsensed information, which includes the time needed for data sensing, face recognition, and data delivery to the Collection Center. So we need to differentiate between the network delay for the data delivery and the time for the recognition process to make a better assessment. Figure 6.4 illustrates the network delay *i.e.*, d_{e2e} [ms], when varying the UAVs/GVs density without face recognition delay. It shows that the use of LTE technology in UAVs and GVs delivers fast speeds, greater bandwidth, and an extremely low E2E delay, which does not go beyond 120 ms. Furthermore, when the Collectors density increases, the delay will become larger due to the higher chances of packet collisions, and vice versa, Figure 6.5 shows the total E2E delay, which is higher due to the constraint of the face recognition process. The targeted monitoring application can however tolerate such a delay, which is lower than 1.2 s even in the worst case.

It should be noted that there are many algorithms and systems available in the literature that provide a higher speed than GaussianFace in the face recognition process, but many of them suffer from accuracy. For instance, when we tested a traditional algorithm for face detection, *i.e.*, the Viola Jones' algorithm [90], which uses the Haar Classifier for face detection, and couples it with a real-time face recognition pattern, using the LBPH algorithm, the average time taken per frame for face recognition is about 34.90 ms. Therefore, we recommend selecting a successful recognition algorithm like GaussianFace to avoid collecting duplicate sensed information, which is a crucial alternative in our design.

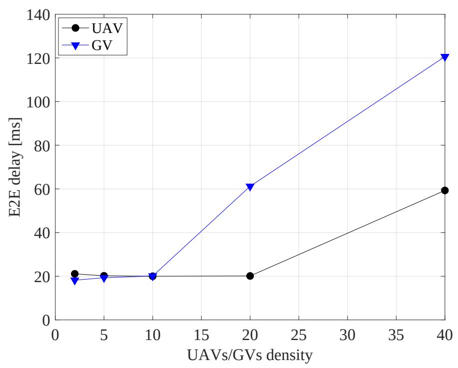


Figure 6.4: E2E (network) delay vs. the UAVs/GVs density.

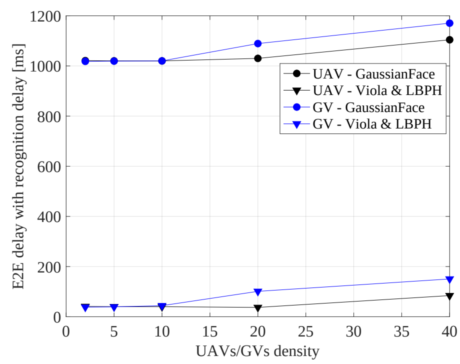


Figure 6.5: Total E2E delay vs. the UAVs/GVs density.

6.4.3 QoS metrics evaluation

Figure 6.6 shows the impact of the UAVs/GVs density on the packet delivery ratio. It can be observed that the proposed model gives a significantly high PDR in the LTE environment, that exceeds 99%, with a preference for UAVs but, as the density increases, it begins to gradually decrease because of collision of the packets.

As shown in Figure 6.7 and Figure 6.8 respectively, other QoS metrics, such as the throughput and jitter. As expected, Figure 6.7 describes how the average throughput Θ in the network increases with the UAVs/GVs density, thanks to the higher number of collected information. The behavior is the same in Figure 6.8, when increasing the vehicles' density, an increasing in mean jitter is observed, due to higher collisions and congestion.

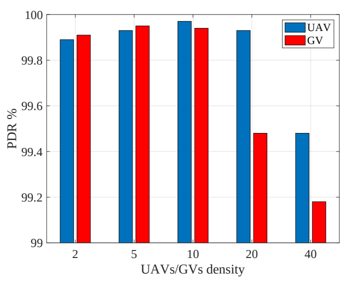


Figure 6.6: PDR variation vs. UAVs/GVs density.

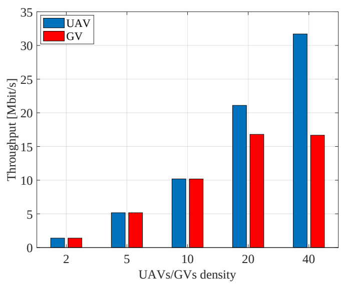


Figure 6.7: Throughput variation vs. UAVs/GVs density.

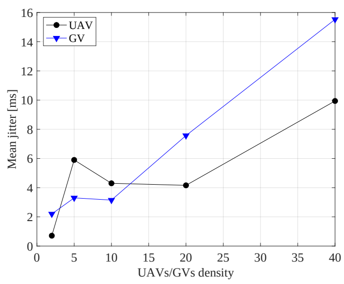


Figure 6.8: Mean jitter variation vs. UAVs/GVs density.

By comparing GUAVA performance with another crowdsensing evaluation present in literature namely BUCST [65], to highlight the benefit of sensing by cooperating between GVs and UAVs, where only UAVs are used in BUCST framework, while GVs are placed for communication backbone. So, we study related cost in terms of traffic overhead and the convergence ratio of both frameworks with respect to the monitored area and the required monitoring time.

The monitoring overhead comparison, depicted in Figure 6.9 between GUAVA and BUCST frameworks shows the advantage of GUAVA over BUCST introduces a reduced overhead compared to the BUCST, due to (i) the cooperation of GVs and UAVs that act both as Collectors and (ii) the face recognition algorithm that avoids the transmission of duplicated data. In addition, the Figure 6.10 represents a graph comparison between BUCST and our proposed approach in terms of monitoring time, the scenario consists of 2 UAVs, 10 GV, and only 1 edge server (*i.e.*, Collection Center) in every 1 km^2. The results obtained prove that the GUAVA framework is superior in terms of the sensor period, as it is able to monitor up to 16 km^2 in about 10 minutes clearly bypassing the BUCST framework that achieved maximum coverage reached 9 km^2 in a similar period of time, which shows the importance and effectiveness of cooperation between GVs and UAVs.

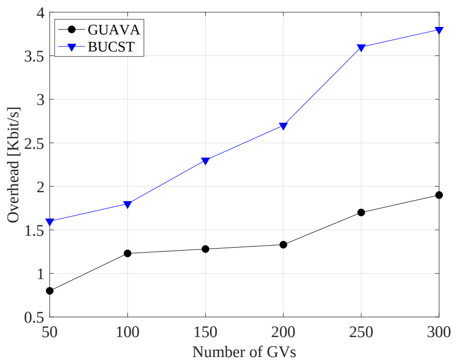


Figure 6.9: GUAVA additional overhead compared to BUCST.

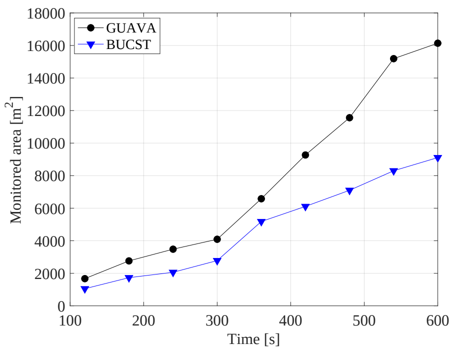


Figure 6.10: GUAVA area monitoring convergence compared to BUCST.

6.5 Discussion

- **Energy consumption of UAVs.** the UAVs are supplied with energy from time to time in order to complete their sensing task, which leads to temporary interruption of the crowdsensing service. MCS mechanism requires some optimization based on the size of the geographic area to be covered, and the approximate number of people to be detected and sensed. To get some seamless, before the UAV battery exceeds its lifetime, a backup UAV can be used to complete the mission during the recharging process or divided the targeted area into smaller regions that are assigned to more UAVs.

- **UAVs Operations Management.** a key challenge is to conduct mission planning for UAVs, and how choosing the optimum flight height to ensure full coverage and increase visibility in the designated area, without interference with other devices and obstacles, addressing changing weather conditions (i.e., strong winds, fogs, etc), as well as to achieve the desired protection when a crash occurs due to the UAV's battery voltage dropping down suddenly or damage occurs, adopting security solutions such as using small parachutes became primordial, that is, to protect pedestrians from potential accidents.

- **Twins prediction.** although the recent face recognition algorithms bypassed human performance in-person identification, they still have challenges associated to detect/identify look-alike faces, like twins, Therefore additional techniques are urgently needed to deal with these similarity issues. Furthermore, Deepfakes techniques that deceive existing algorithms [91] can also spread, so it is necessary to improve Deepfake Detection Algorithms and solve this issue related to look-alike faces or adopt a hybrid system that associates face recognition technique with upper-body recognition [92] (e.g., shoulder-to-shoulder width, neck length/width, chest/waist size and back length, etc.), which allows giving more accuracy.

- **selecting targeted Geo-areas smartly.** to avoid random sensing of different regions according to the evolution of the pandemic, GUAVA typically targets places of mass gathering or places where the infection rate is expected to be high (*ie*, the so-called red areas), along with other relevant information that can achieve maximum benefit from the use of UAVs, such as the expected number of people to be sensed and the expected weather conditions.

6.6 Concluding Remarks

In this chapter, we present a crowdsensing solution namely GUAVA, leveraging both ground and flying vehicles, that combines thermal cameras, GPS devices, and an edge server, by tracking the infected areas and thus containing the COVID-19 pandemic. The task of vital signs sensing of pedestrians in urban areas is supported by the AI through a face recognition algorithm, to discard duplicate sensed parameters with high accuracy,

Once the gathered data was uploaded to the edge server, data processing, and visualization techniques are applied for creating a detailed heat-map, which paves the way for early decision support and intervention, in addition to smart search (e.g., locate lost children), between key stakeholders including police and health dept.

Finally, we evaluate the effectiveness and usefulness of our proposal in terms of QoS metrics, considering the trade-off between the accuracy of the face recognition and E2E delay. Furthermore, we compared GUAVA area monitoring convergence against BUCST approach, results show that GUAVA offers significantly better performance at reducing monitoring overhead. As a future work, we can focus on the aforementioned challenges, and build a prototype with a small set of collectors, in addition to enhancing the facial recognition algorithm with other algorithms such as gait recognition to address the similarity issues.

"True wisdom consists of ***tracing*** *effects to their causes."*

— Oliver Goldsmith

Chapter 7

TraceMe: Contact Tracing Framework based on OSN for Early Prevention of Infectious Disease Outbreaks.

Contents

7.1 Introduction **76**
7.2 Related Work **77**
7.3 TraceMe Model **80**
7.4 Evaluation **84**
7.5 Concluding Remarks **89**

7.1 Introduction

Unlike other solutions that rely in IoV approach to break the chain of spread of infection among people, presented in the previous chapters(4, 5 & 6). Contact Tracing (CT) has emerged as the easiest and least costly solution to identify people exposed to Covid-19. The slow process for identifying suspected cases is an important concern for manual CT, and with the advent of smart devices, such as smartphone, smartwatch, etc, equipped with wireless communication technologies, like LTE, WiFi and Bluetooth, and localization systems, like GPS, led to the emergence of digital tracing applications, so-called " Digital Contact Tracing (DCT)", and allows to implement proximity approaches. However, most of these CT solutions are typically implemented as user-dependent smartphone applications that raise multiple privacy concerns among citizens.

In this chapter, for the CT task that does not require a user-dependent smartphone applications, we design a DCT framework, namely TraceMe, leveraging Mobile Wireless Networks (MWNs) to identify physical contacts, coupled with Online Social Networks (OSNs), which provide a successful way to track, share and exchange information in real-time at low cost, as illustrated in Figure 7.1. Specifically,TraceMe is designed based on the following contributions:

- To the best of our knowledge, this is the first work that aims to detect and prevent patients with COVID-19 disease in real time through digital CT, by leveraging Mobile Wireless Networks(MWNs) coupled with OSNs, , in order to (i) detect and monitor infections through "physical" proximity contacts, and (ii) identify exposed people via "virtual" social proximity through OSNs, (Section 7.3);

- We evaluate TraceMe with a realistic dataset - Stanford Network Analysis Project (SNAP) [93]. The results show the high capability of TraceMe to detect communities of close contacts at high risk of infection in large-scale social networks, (Section 7.4);

- We consider a popular theoretical epidemic evolution model, namely the Susceptible-Exposed-Infectious-Removed (SEIR) model [94], to investigate the forecast of how this solution can mitigate the infection curve, (Section **??**).

7.2 Related Work

7.2.1 IoT based-CT apps

IoT network has gained more attention in recent years, and IoT-based CT depends on sensing technologies with the analysis mechanism, which evoked increasing concentrations. Many studies are now preoccupied with applying CT techniques based on IoT devices. different strategies with heterogeneous IoT devices have been put forward, relying on contact-based or contact-less sensing.

The current applications mainly focus to trace contacts in the early stage based on proximity or location traces, using various devices (e.g., thermal cameras, GPS, Bluetooth, WiFi, etc.).

TraceTogether [95], is a centralized contact tracing approach implemented in Singapore, leveraging BLE-based mobile phones, to battle COVID-19 through identifying if two persons who are in close contact (within about 6 feet of each other) , they generate and share their contact tokens with The authorities database, the later uses phone contacts to alert users about potential exposure to Covid-19 if a token matches an infected person's token in the database.Around the same context, the use of Bluetooth for proximity detection was also used in Covid Watch [96] and COVIDsafe [97] , automated and anonymous contact tracing apps, that uses Bluetooth signals to identify the proximity of users to an infected person and sends alerts about possible exposures, and guide them to take further measures. GoCoronaGo (GCG) [98], is another contact tracing application that use BLE signals scanning periodically, to discover and collect nearby devices to the infected user, in addition to allowing users to share their GPS location voluntarily in the back-end, to increase the reliability of detection, and then build a temporal contact graph of devices by the health center, where the proximity considered as a relationship between them, to identify and alert primary and secondary contacts.

The authors in [99], [37] discuss the use of thermal cameras with GPS devices embedded on the ground or flying vehicles to identify infected persons based on their two vital signs (body temperature, breathing rate), and localize them, the sensed data is collected in an edge-based database for COVID-19, hence, creating a heat-map of infected areas to take prior measures. M. Patel in [100], Developed a GPS-based CT application to fight against the COVID-19, that relays on citizens either infected or healthy, as well as various government agencies including Municipal Corporations, Hospitals, Police Departments, Health Departments, and non-governmental organizations (NGOs), etc, to work in cooperation, where they share the location of every infected or suspected person with a central government server. The latter allows sharing with the clients' apps an ArcGIS Dashboard by presenting location-based analytic to enable visualization on the map infectious areas in real-time, but privacy remains one of the biggest issues.

7.2.2 Applying SNA for CT

X. Liao *et al.* in [101], propose a design that relies on Tripartite Graph Clustering for communities detection of Twitter users, based on coronavirus-related pandemic keywords, and sentiments analysis, focusing on English-language tweets. R. Goel *et al.* in [102],study the use of highly influential users' or "leaders" tweets during the Covid-19 pandemic, which could be individual users or organization like WHO, using network, text, and sentiment analysis, then classify them into four different communities: research, news, health, and politics, based on their general interests(e.g, in politics, they tend to share keywords about travel and hygiene, while in research, they tend to share symptoms, development of vaccination keywords), hence, a machine learning algorithm is used to build a prediction model for community detection.However, this model cannot be generalized to other pandemic such as Ebola. The authors in [103], attempted to predict the future number of influenza patients in UAE, and developed a context-aware classifier called repTweets, to classify influenza-related tweets for both Arabic and English into non-reporting and reporting tweets . The latter is classified to self and non-self reporting. The great value of Normalized Cross Correlation (NCC = 0.78 on average) gave the architecture high reliability. In the same context the authors in [104], extract data from English and Filipino tweets, related to COVID-19 vaccination in the Philippines, and classify them into positive, neutral, and negative opinions via using a Naïve Bayes model with 81.77% accuracy, to enable the government to find an effective strategy regarding financial cover, vaccination provision, and scheduling. K. Cresswell *et al.* in [105], leverage AI-enabled Twitter and Facebook posts analysis, to classify public attitudes (positive, neutral, negative) toward contact tracing apps in the United Kingdom, which facilitate carrying out effective public health strategies.

7.3 TraceMe Model

as shown in Figure7.1, we propose TraceMe , a contact tracing framework, which we model as a two-stage process. The first stage consists of real-time and high-precision detection of potentially infected people using proximity tracking tools implemented by means of traditional MWNs. Then, in the second stage, we exploited the information about physical contacts to identify more potentially exposed contacts to Covid-19 based on OSNs using Social Network Analysis (SNA) tools for community detection. So as illustrated in Fig 1. we consider two network layers, i.e., (i) a physical layer based on MWNs, and (ii) a virtual layer based on OSNs. Every node in the physical layer has a corresponding node in the virtual layer, and each virtual node and each node(representing the infected person) has links with its close contacts in the same community and links with the other communities. It should be noted that in TraceMe, if any node in the MWN has no corresponds in the virtual layer (ONS), we used only traditional proximity-based tracing approaches. The TraceMe framework contains two key components:

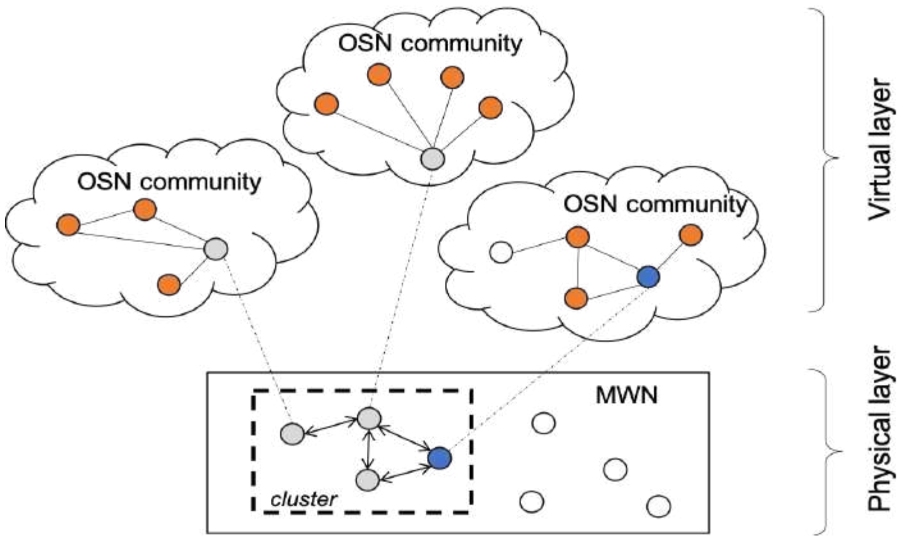


Figure 7.1: Overview of TraceMe approach based on MWNs and OSNs.

7.3.1 MWN-based contact tracing

In the physical layer, TraceMe performs a proximity-based contact tracing, when a confirmed case of Covid-19 is detected, by the health authorities at the moment of testing (i.e., the blue node in Figure7.1), it is possible to determine its location and movements by measuring the power level coming from its personal device, i.e., a smartphone, similarly to [106]. Therefore, people that are in close contact with a COVID-19 case (i.e., the grey nodes in the cluster of Figure7.1), will receive mass and urgent SMS messages notification on their mobile devices, about the possibility of being exposed to the virus and the necessity of taking some precautions such as physical distancing or quarantine.

The data relating to the Covid-19 cases, i.e., name, social network profile(s), address, and phone number are cached in a secure Health Department Database that will be used for the second stage to trace more close contacts through TraceMe framework. We rely on state-of-the-art encryption and authentication techniques [107], for privacy preservation, confidentiality, and integrity of the Health Department Database. The infected person is removed automatically from the Database when the state of the test changes from positive to negative.

7.3.2 OSN-based contact tracing

This stage runs on every infected person that has a Linkage to his user account across the social platform. Indeed, OSNs are among the tools that create new relationships, through the dissemination and sharing of information of common interest, in addition to the great ease of use even among the elderly, where when identifying infected persons at the physical layer (MWNs), we can discover other cases at exposed to the risk of infection, through the use of SNA tools.

The OSN profile(s) of infected people cached in the Health Department Database, can be used in TraceMe for community detection, and to identify the list of contacts (i.e., 1-degree neighbors), with different levels of social ties (i.e., strong and weak relationships). Contact tracing that TraceMe uses to identify people who have been exposed to someone with an infectious are close contacts of a user (i.e., orange nodes in Figure7.1)), leveraging the information that creates strong social ties, according to some extracted features. Once a set of close contacts (i.e., a clique) is detected, TraceMe informs them about the potential exposure to the disease by

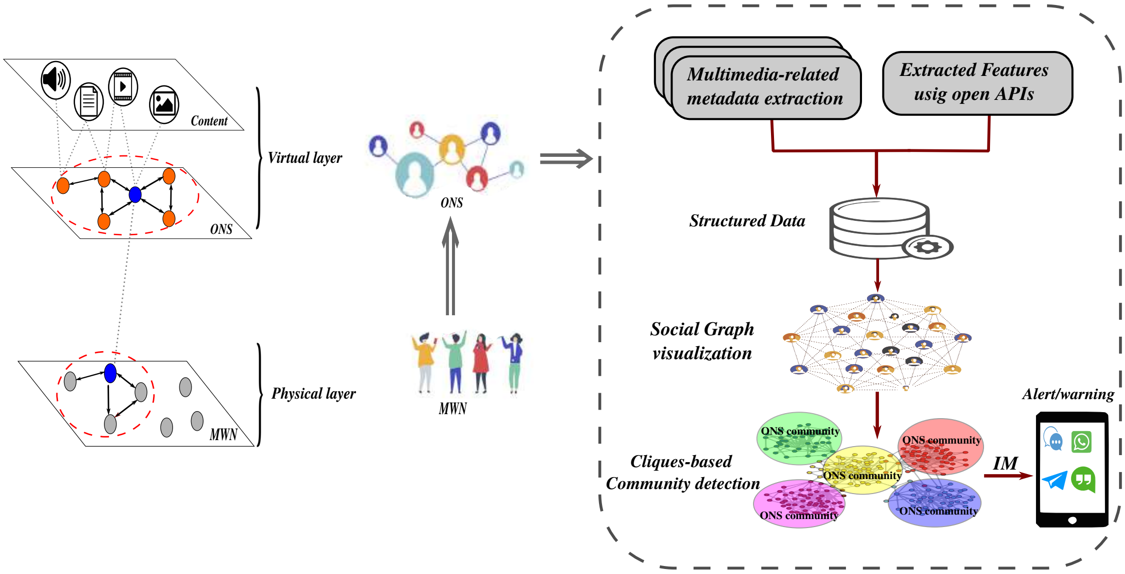


Figure 7.2: Cliques-based community detection in TraceMe to identify and notify close contacts.

means of text/instant messaging tools (like Messenger on Facebook). Privacy is preserved in TraceMe, which is based on the notification about the likely exposure to the disease without identifying the infected person.

Now we explain in a little bit detail TraceMe's workflow during a contact tracing task in the ONS layer stage, as shown in Figure 7.2:

- **Data collection and extraction.** only authorized health staff can access the personal and medical information stored in the database, and use the social profile links of the infected persons. A technique of extracting the features (e.g., visited location, closest friends, attended events, etc) is useful using open APIs, In addition, TraceMe can leverage the open-source Metadata Extraction Tools, to automatically extract location-related descriptive information, so-called metadata from a range of digital file formats including private Videos/Audios/Images files, PDF/DOCX documents, and many other private files, so the Geo-location of users are represented in the shared files' metadata, which includes more precise details about physical location information such as country, city, latitude, and longitude, then, TraceMe applies a data cleaning technique to remove the replicate data, incorrect, and corrupted ones, hence, structuring raw data.

- **Graph representation.** a social network graph is created based on the extracted features in the previous stage are exploited, so, $G(V, E)$, comprised of $V = \{v_1, v_2, \ldots, v_N\}$, and $E = \{e_1, e_2, \ldots, e_M\}$, as the set of vertices and edges, respectively. The SNA used a social network graph, so-called a sociogram, for a visual representation, for highlighting the social ties between the infected user and his close contacts. Indeed, each vertex represents a user, while each edge represents the social relationship between two users.

- **Clique discovery and community detection.** TraceMe applies a clique-based mechanism on the sociogram to highlight the closest ties, so we choose Clique based community detection algorithm among other algorithms, for its high performance in terms of accuracy and low computational complexity [108].

  More specifically, given a graph $G(V, E)$, a clique $G_1(V_1, E_1) \in G$ is a maximal complete sub-graph, *i.e.*:

  $$\forall\, e_{ij} = (v_i, v_j) \in E_1, \implies v_i, v_j \in V_1. \tag{7.1}$$

  This indicates that all the vertices in G_1 are completely connected with each other. In our strategy, a clique is basically a group of users where everybody is connected directly to everyone else. The clique-based community detection method proceeds by splitting the social network graph into disconnected complete sub-graphs and then groups the adjacent cliques. Then, it merges two cliques together if one clique of m nodes has $(m - 1)$ nodes in common with another clique, in order to form the final community structure. Therefore, the Clique-based community detection algorithm in TraceMe helps to find strongly connected users and to analyze their behavior in order to build an ideal graph based on the commonly shared features.

  TraceMe allows showing different sociograms per each infected user, leveraging his input features, and as a result, we can see different cliques. For instance, given an infected user u, TraceMe can identify as candidate close contacts the friends of u that attended the same event in a given time window (e.g., the last

10 days), or extract multimedia content (i.e., photos and videos) shared by user u in specific a period of time to identify his close contacts friends, as well as, Trace me can identify contacts of u that registered in the same locations. Indeed, several OSN platforms today use check-in features, e.g, Foursquare, Facebook, Gowalla, etc., to help share users' locations. which allows following human mobility, infer social ties, and understanding community behavior and the infection trend.

7.4 Evaluation

To evaluate the real-world applicability of TraceMe, we use realistic social network dataset built according to the check-ins shared by users. The framework is implemented using Python in the Google Colab environment.

7.4.1 Description of Dataset

Gowalla check-in [93] is the dataset used in this study. Gowalla[1] is an American location-based social networking service, which allows users to report their presence in a specific place by check-in process. the Gowalla dataset has been compiled using public APIs, within the Stanford Network Analysis Project (SNAP), containing 196,591 nodes and 950,327 edges. The variables in this dataset are user ID, check-in time, latitude, longitude, and the location ID from the period of Feb. 2009 till Oct. 2010.

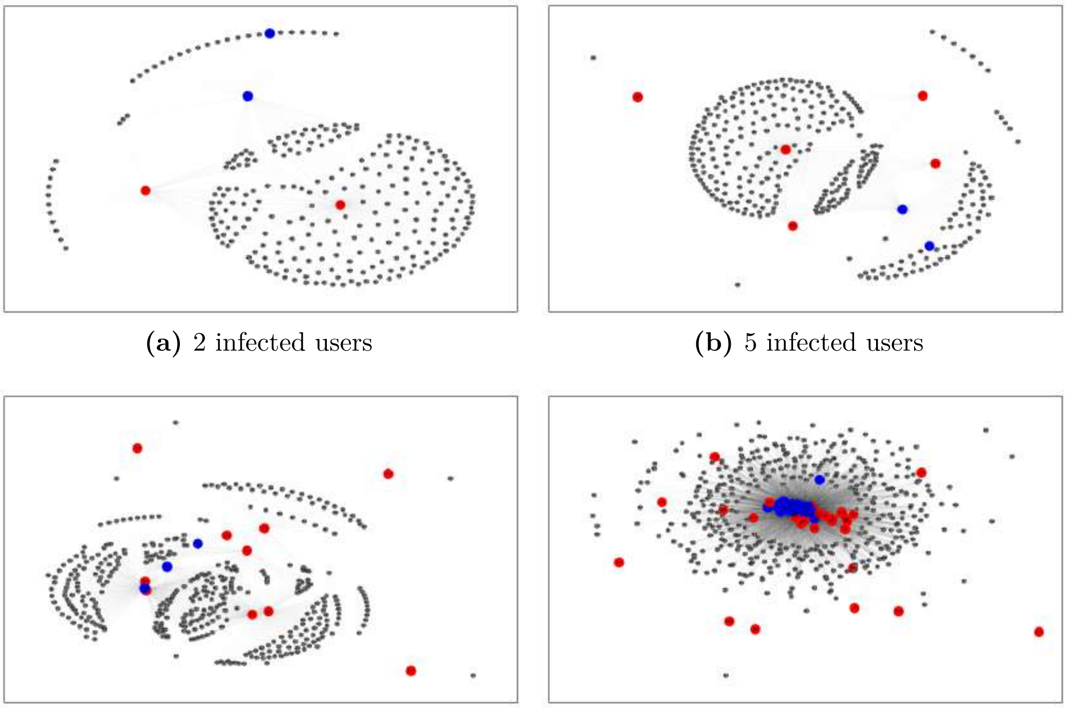
7.4.2 Social network analysis

The popular python NetworkX package is for social network construction, analysis, and creating a graph G(V, E), where ties E(edges) means that the linked users have some type of interaction in the community. Users share their geographical location information, while a direct link from V_a to V_b, means that the two users a and b have visited the same Geo-location.

[1] https://go.gowalla.com

Table 1: Users at risk according to the number of infected users.

Infected users	Users at risk	User at high risk
2	271	2
5	317	2
10	378	3
50	463	23



(a) 2 infected users **(b)** 5 infected users

(c) 10 infected users **(d)** 50 infected users

Figure 7.3: Users at high risk of infection (*blue nodes*) when varying the number of infected users (*red nodes*).

Our analysis was performed in 2 steps. First, we build an undirected graph using networkx from the edge list after removing duplicate tuples, then visualize the social ties between the users. The relationship(edge) is created between every pair of users if they have visited the same location ID at the same or closest time. The social graph is a bit more readable and gives us a brief overview of the data.

Second, we use clique-based community detection to discover the people at risk of the pandemic, we use clique because analyzing part of the clique enables us to predict the rest of the clique members. To evaluate our design, TraceMe adopted the Bron–Kerbosch algorithm [109], to discover all maximal cliques in the undirected social graph. By randomly varying the number of infected users from the Gowalla users dataset from 2 to 50, listed in Table I. The table shows the obtained users at risk by considering the co-location for each number of infected users, as well as the number of highly infected users, which is largely reduced to ensure more accuracy, via applying a community detection algorithm to find and by merging cliques. Figure 7.3 shows the visual representations of Table 1 results, which help us to understand data quickly and gained clarity for the right decision-making.

7.4.3 A theoretical Forecasting using the SEIR model

To assess TraceMe in terms of infection evolution, we implement a mathematical model namely the SEIR model [94], which is a deterministic compartment epidemic model, that is widely used to represent an assembly of four parts of population-based on disease stages as follows: the susceptible s(t), the exposed e(t), the infected i(t), and the recovered r(t) population size according to time t. The generalized SEIR model considers a system of four differential equations, as follows:

$$\frac{dS}{dt} = -\frac{\beta SI}{M} \tag{7.2}$$

$$\frac{dE}{dt} = \frac{\beta SI}{M} - \alpha E \tag{7.3}$$

$$\frac{dI}{dt} = \frac{\beta SI}{M} - \gamma I \tag{7.4}$$

$$\frac{dR}{dt} = \gamma I \tag{7.5}$$

where β, α, and $\gamma \in [0, 1]$ are model parameters influencing the transition between the different stages.

The progression of the pandemic can be modeled by the diagram of SEIR compartmental model, as depicted in Figure7.4, using the following rates:

- $(1 - m^{ct})\frac{\beta SI}{M}$: is the rate at which the susceptible population becomes exposed to infection, that will lead to the disease outbreaks, where $m^{ct} = m^{ct}_{mwn} + m^{ct}_{ons}$,

m^{ct} describes the effect of using both MWN and ONS contact tracing strategy for notification and take further measures, if $m^{ct}_{ons} = 0$ means no CT-based ONS measures.

- αE: is the rate at which exposed population becomes ineffective.
- γI: is the rate at which the infected population recovers from infection.

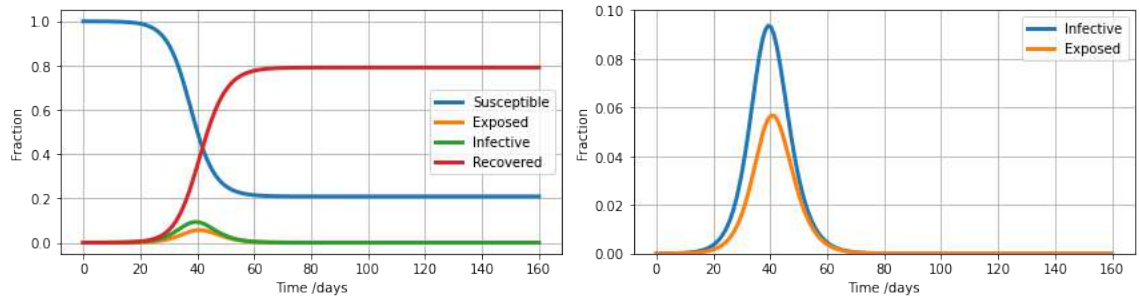
β, α, γ are the model parameters. This results in four equations:

$$\frac{ds}{dt} = -(1 - m^{ct})\beta si \tag{7.6}$$
$$\frac{de}{dt} = (1 - m^{ct})\beta si - \alpha e \tag{7.7}$$
$$\frac{di}{dt} = \alpha e - \gamma i \tag{7.8}$$
$$\frac{dr}{dt} = \gamma i \tag{7.9}$$

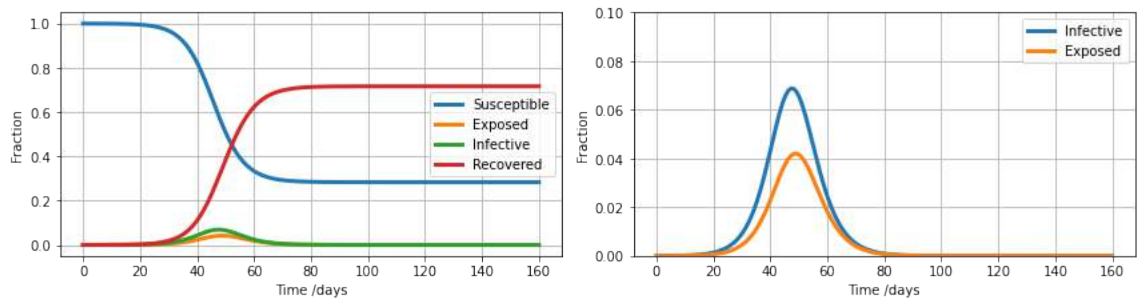
we assume that m^{ct}_{mwn} is at a fixed rate, we observe that the fraction of infected population and those who are exposed to the disease is decreasing while increasing the rate of use of ONS-based CT, as shown in Figure7.5, and this is due to the exposure notification system that TraceMe used, and the impact of the taken measures (e.g. physical distancing, quarantine, etc) to limit disease transmission.

$$\boxed{\mathrm{S}} \xrightarrow{\frac{(1-m^{ct})\beta SI}{M}} \boxed{\mathrm{E}} \xrightarrow{\alpha E} \boxed{\mathrm{I}} \xrightarrow{\gamma I} \boxed{\mathrm{R}}$$

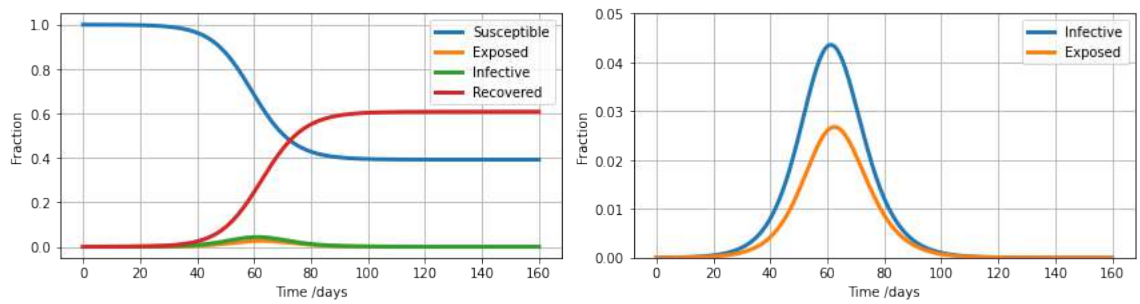
Figure 7.4: The diagram of the SEIR compartmental model



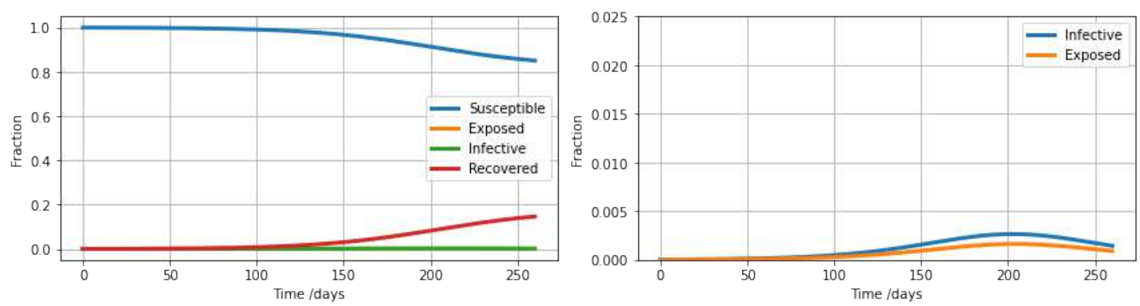
(a)



(b)



(c)



(d)

Figure 7.5: Modeling pandemic outbreak with respect to time by varying the rate the use of ONS, in Figure7.5a: $m_{ons}^{ct} = 0$, in Figure7.5b: $m_{ons}^{ct} = 0.1$, in Figure7.5c: $m_{ons}^{ct} = 0.2$, in Figure7.5a: $m_{ons}^{ct} = 0.4$.

7.5 Concluding Remarks

In this chapter, we attempt to limit the spread of the pandemic by considering the correlation between physical infected users through MSNs and their OSN accounts, to identify the close contacts.To that end, we propose a novel Contact Tracing framework, namely TraceMe, which is able to discover tight communities of contacts that are at high risk of infection using the notion of cliques, through the use of the traditional physical proximity detection relying on MWNs, with OSNs and social network analysis. A theoretical trend supported by a realistic performance evaluation based on the Gowalla dataset shown a high effectiveness of this design, especially when the number of personal multimedia contents shared by users are increased(e.g., images, videos, etc.), which are useful for contact-detection using their extracted time-space information. However, in some cases, such multimedia files lack metadata information, which hampers localizing them, thus prompts thinking about alternative ways for contact tracing, such as the use of deep learning algorithms.

““In literature and in life we ultimately pursue, not conclusions, but beginnings.””

— Sam Tanenhaus.

Chapter 8

Conclusion, Future Work and Publications.

Contents

8.1 Contributions summary 90
8.2 Future Research Opportunities 92
8.3 Publication list 92

As a new approach and alternative way to replace traditional Mobile Crowdsensing paradigms, Vehicular Crowdsensing provides a cost-effective, no energy dependent solution for large scale sensing, especially in the era of the pandemic, which allowed to full coverage in real-time without budget, the matter that help to mitigate or stop its outbreak. This dissertation, following this research direction, attempts to address some of the research issues related to computational power processing, energy consumption, and achieving efficient information quality trade-off for decision maker. In this final chapter, our main contributions are summarized, followed by a brief discussion about future research opportunities.

8.1 Contributions summary

The main contributions of this dissertation focuses on the use of VCS, reinforced with edge computing to improve the power of processing and provide clear vision for intervention to contain the spread of COVID-19 at large scale in real-time, so vehicles participate in the sensing tasks, then, uploading sensed data at edge computing level.

Remote sensing framework-based VCS: In Chapter 4, we proposed a new design for pandemic monitoring like Covid-19 that combines vehicles equipped with sensors and GPS devices, and enhanced with edge cloud, to identify and Tracking infected pedestrians via sensing their vital signs. In detail, we used thermal cameras with GPS trackers, to measure the body temperature and breathing rate of pedestrians, then, localize infected areas. edge computing required for data processing and visual representation(Geo-heatmap), which provides real-time and perfect information pictures about the spread of the disease for early intervention. Finally, we evaluate our proposal, which shows that it is able to provide high performance in terms of QoS metrics

DeepDist: In Chapter 5, a physical distancing notification system based on deep learning (Faster R-CNN algorithm) and the Internet of Vehicles, digital advertising boards (for pedestrians), and OBU(for vehicles) can be leveraged to show/make violation notifications, since each vehicle is equipped with thermal and vision cameras, to detect objects and measure the distance between objects of the same class. This design is called DeepDist, which also can be used by Cloud-based services, to take prior knowledge about crowded places.

GUAVA: In Chapter 6, a real-time and full coverage MCS design is implemented for urban non-urban environment monitoring applications, as the sensing tasks can reach the muddy, unpaved, and difficult-to-reach places, etc. The proposed design is called GUAVA, which is based on cooperative sensing between UAVs and GVs, with the use of an artificial intelligence algorithm(gaussianFace) to recognize faces and avoid duplication in sensed data. a wireless power charging for UAVs is implemented to ensure sensing tasks quality. This model is an improvement to the design mentioned in Chapter 4, as the evaluation results have proven its superiority in terms of Qos metrics and its high faces recognition rate.

TraceMe: In Chapter 7, a real-time contact tracing design is implemented, leveraging OSNs and social network analysis, in addition to using traditional physical proximity detection relying on MWNs, coupled with OSNs and social network analysis to identify more cases exposed to Covid-19, by applying cliques-based community detection algorithm.

8.2 Future Research Opportunities

This ph.D thesis proposes three novel VCS designs to help mitigate or stop the spread of the Covid-19 pandemic, While several key issues have been investigated such as duplication, privacy, etc, various research opportunities exist for improving our models are considered as future works.

Malicious Vehicles'. either from drivers using federated vehicles(police), or other vehicles that use some form of attacks such as phishing attacks [38], which brings security violations in sensing tasks, thus uploading faked sensed data. Following a security policy is a key matter, such as encryption, machine/deep learning algorithm.

Multiple Task allocation. indeed, these MCS frameworks integrate smart cameras Sensors, that are capable to acquire a large amount of diverse data in one sensing campaign (e.g. body signs, identifying people, traffic monitoring, etc), which promotes the sensing coverage and data diversity for use it in different aspects and help to make better decisions.

The goal of the thesis has been reached after all objectives have been achieved, So we can now conclude this ph.D thesis. The last section of this chapter enumerates the related publications.

8.3 Publication list

This section lists the publications related to this ph.D thesis, in addition to other collaborations published during this period of time.

Conference publications

1. **Y. Sahraoui**, L. De Lucia, A. M. Vegni, C. A. Kerrache, M. Amadeo, and A. Korichi, "TraceMe: Real-Time Contact Tracing and Early Prevention of COVID-19 based on Online Social Networks," *in 2022 IEEE 19th Annual Consumer Communications Networking Conference (CCNC), 2022, pp. 893–896.*

2. **Y. Sahraoui**, C. A. Kerrache, A. Korichi, A. M. Vegni, and M. Amadeo, "LearnPhi: a Real-Time Learning Model for Early Prediction of Phishing Attacks in IoV," *in 2022 IEEE 19th Annual Consumer Communications Networking Conference (CCNC), 2022, pp. 252–255.*

Journal publications

1. **Y. Sahraoui,** A. Korichi, C. A. Kerrache, M. Bilal, and M. Amadeo, "Remote sensing to control respiratory viral diseases outbreaks using Internet of Vehicles," *Trans. Emerg. Telecommun. Technol., p. e4118, 2020.*

2. **Y. Sahraoui**, C. A. Kerrache, A. Korichi, B. Nour, A. Adnane, and R. Hussain, "DeepDist: a deep-learning-based IoV framework for real-time objects and distance violation detection," *IEEE Internet Things Mag., vol. 3, no. 3, pp. 30–34, 2020.*

3. **Y. Sahraoui**, C. A. Kerrache, and M. Amadeo., A. M. Vegni, A. Korichi, J. Nebhen, and I. Muhammad, "A cooperative crowdsensing system based on flying and ground vehicles to control respiratory viral disease outbreaks," *Ad Hoc Networks, vol. 124, p. 102699, 2022.*

References

[1] Roberto Minerva, Abyi Biru, and Domenico Rotondi. "Towards a definition of the Internet of Things (IoT)." In: *IEEE Internet Initiative* 1.1 (2015), pp. 1–86.

[2] Chen Jiacheng et al. "Software defined Internet of vehicles: Architecture, challenges and solutions." In: (2016).

[3] Jun Zhang and Khaled B Letaief. "Mobile edge intelligence and computing for the internet of vehicles." In: *Proceedings of the IEEE* 108.2 (2019), pp. 246–261.

[4] Wenchao Xu et al. "Internet of vehicles in big data era." In: *IEEE/CAA Journal of Automatica Sinica* 5.1 (2017), pp. 19–35.

[5] Talal Ashraf Butt et al. "Privacy management in social internet of vehicles: review, challenges and blockchain based solutions." In: *IEEE Access* 7 (2019), pp. 79694–79713.

[6] Juan Contreras-Castillo, Sherali Zeadally, and Juan Antonio Guerrero Ibáñez. "A seven-layered model architecture for Internet of Vehicles." In: *Journal of Information and Telecommunication* 1.1 (2017), pp. 4–22.

[7] Kiran Ahuja, Brahmjit Singh, and Rajesh Khanna. "Network selection algorithm based on link quality parameters for heterogeneous wireless networks." In: *Optik* 125.14 (2014), pp. 3657–3662.

[8] Ding Zhang et al. "An access selection algorithm based on GRA integrated with FAHP and entropy weight in hybrid wireless environment." In: *2013 7th International conference on application of information and communication technologies*. IEEE. 2013, pp. 1–5.

[9] Surbhi Sharma and Baijnath Kaushik. "A survey on internet of vehicles: Applications, security issues & solutions." In: *Vehicular Communications* 20 (2019), p. 100182.

[10] Ahmad Yusri Dak, Saadiah Yahya, and Murizah Kassim. "A literature survey on security challenges in VANETs." In: *International Journal of Computer Theory and Engineering* 4.6 (2012), p. 1007.

[11] Jiangtao Wang et al. "Task allocation in mobile crowd sensing: State-of-the-art and future opportunities." In: *IEEE Internet of Things journal* 5.5 (2018), pp. 3747–3757.

[12] Yanan Wang, Guodong Sun, and Xingjian Ding. “Coverage-Balancing User Selection in Mobile Crowd Sensing with Budget Constraint.” In: *Sensors* 19.10 (2019), p. 2371.

[13] Wei Gong, Baoxian Zhang, and Cheng Li. “Task assignment in mobile crowdsensing: Present and future directions.” In: *IEEE network* 32.4 (2018), pp. 100–107.

[14] Nicholas D Lane et al. “Piggyback crowdsensing (pcs) energy efficient crowdsourcing of mobile sensor data by exploiting smartphone app opportunities.” In: *Proceedings of the 11th ACM Conference on Embedded Networked Sensor Systems*. 2013, pp. 1–14.

[15] Robert Ighodaro Ogie. “Adopting incentive mechanisms for large-scale participation in mobile crowdsensing: from literature review to a conceptual framework.” In: *Human-centric computing and information sciences* 6.1 (2016), pp. 1–31.

[16] Luis G Jaimes, Idalides Vergara-Laurens, and Alireza Chakeri. “SPREAD, a crowd sensing incentive mechanism to acquire better representative samples.” In: *2014 IEEE International Conference on Pervasive Computing and Communication Workshops (PERCOM WORKSHOPS)*. IEEE. 2014, pp. 92–97.

[17] Yufeng Wang et al. “QuaCentive: a quality-aware incentive mechanism in mobile crowdsourced sensing (MCS).” In: *The Journal of Supercomputing* 72.8 (2016), pp. 2924–2941.

[18] Delphine Christin, Carsten Büttner, and Nicolas Repp. “CachedSensing: Exploring and documenting the environment as a treasure hunt.” In: *37th Annual IEEE Conference on Local Computer Networks-Workshops*. IEEE. 2012, pp. 973–981.

[19] Anna Dorfman and Igor Grossmann. “Social Incentives.” In: *Encyclopedia of Personality and Individual Differences*. Ed. by Virgil Zeigler-Hill and Todd K. Shackelford. Cham: Springer International Publishing, 2020, pp. 5067–5070. URL: `https://doi.org/10.1007/978-3-319-24612-3_1834`.

[20] Xiaowen Gong et al. “Exploiting social trust assisted reciprocity (STAR) toward utility-optimal socially-aware crowdsensing.” In: *IEEE Transactions on Signal and Information Processing over Networks* 1.3 (2015), pp. 195–208.

[21] Djallel Eddine Boubiche et al. “Mobile crowd sensing–Taxonomy, applications, challenges, and solutions.” In: *Computers in Human Behavior* 101 (2019), pp. 352–370.

[22] Liang Liu et al. “Third-eye: A mobilephone-enabled crowdsensing system for air quality monitoring.” In: *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 2.1 (2018), pp. 1–26.

[23] Faria Kalim, Jaehoon Paul Jeong, and Muhammad U Ilyas. “CRATER: A crowd sensing application to estimate road conditions.” In: *IEEE Access* 4 (2016), pp. 8317–8326.

[24] Ilsun You, Kim-Kwang Raymond Choo, Chi-Lun Ho, et al. “A smartphone-based wearable sensors for monitoring real-time physiological data.” In: *Computers & Electrical Engineering* 65 (2018), pp. 376–392.

[25] Lampros A Kalogiros et al. “Allergymap: a hybrid mHealth mobile crowdsensing system for allergic diseases epidemiology: a multidisciplinary case study.” In: *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*. IEEE. 2018, pp. 597–602.

[26] Cheri A Levinson, Laura Fewell, and Leigh C Brosof. “My Fitness Pal calorie tracker usage in the eating disorders.” In: *Eating behaviors* 27 (2017), pp. 14–16.

[27] Syed R Rizvi, Susan Zehra, and Stephan Olariu. “Aspire: An agent-oriented smart parking recommendation system for smart cities.” In: *IEEE Intelligent Transportation Systems Magazine* 11.4 (2018), pp. 48–61.

[28] S Sanchana, AP Manjari, and Bala Naga Jyothi. “Application Development for Real-Time Location Tracking for Underwater Vehicles Using Low-Cost GPS with GSM.” In: *2021 Sixth International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*. IEEE. 2021, pp. 108–112.

[29] Raghu K Ganti, Fan Ye, and Hui Lei. “Mobile crowdsensing: current state and future challenges.” In: *IEEE communications Magazine* 49.11 (2011), pp. 32–39.

[30] Wei Feng et al. “A survey on security, privacy, and trust in mobile crowdsourcing.” In: *IEEE Internet of Things Journal* 5.4 (2017), pp. 2971–2992.

[31] Yuanni Liu et al. “Reverse auction based incentive mechanism for location-aware sensing in mobile crowd sensing.” In: *2018 IEEE International Conference on Communications (ICC)*. IEEE. 2018, pp. 1–6.

[32] Chatschik Bisdikian, Lance M Kaplan, and Mani B Srivastava. “On the quality and value of information in sensor networks.” In: *ACM Transactions on Sensor Networks (TOSN)* 9.4 (2013), pp. 1–26.

[33] Francesco Restuccia et al. “Quality of information in mobile crowdsensing: Survey and research challenges.” In: *ACM Transactions on Sensor Networks (TOSN)* 13.4 (2017), pp. 1–43.

[34] Jianbing Ni et al. “Security, privacy, and fairness in fog-based vehicular crowdsensing.” In: *IEEE Communications Magazine* 55.6 (2017), pp. 146–152.

[35] Lewis Nkenyereye et al. “Secure crowd-sensing protocol for fog-based vehicular cloud.” In: *Future Generation Computer Systems* (2021).

[36] Adarsh Kumar et al. “A drone-based networked system and methods for combating coronavirus disease (COVID-19) pandemic.” In: *Future Generation Computer Systems* 115 (2021), pp. 1–19.

[37] Yesin Sahraoui et al. “Remote sensing to control respiratory viral diseases outbreaks using Internet of Vehicles.” In: *Transactions on Emerging Telecommunications Technologies* (2020), e4118.

[38] Yesin Sahraoui et al. “DeepDist: A Deep-Learning-Based IoV Framework for Real-Time Objects and Distance Violation Detection.” In: *IEEE Internet of Things Magazine* 3.3 (2020), pp. 30–34.

[39] Jose M Cecilia et al. “WATERSensing: A smart warning system for natural disasters in Spain.” In: *IEEE Consumer Electronics Magazine* (2021).

[40] YIN Yuehong et al. “The internet of things in healthcare: An overview.” In: *Journal of Industrial Information Integration* 1 (2016), pp. 3–13.

[41] Fayez Hussain Alqahtani. “The application of the Internet of Things in healthcare.” In: *Int. J. Comput. Appl* 180.18 (2018), pp. 19–23.

[42] Atrayee Gupta, Sayantani Saha, and Sriyanjana Adhikary. “Delay-sensitive remote health framework for secured wban using sensor virtualization.” In: *Applications and Innovations in Mobile Computing (AIMoC)* (2016).

[43] Jeremy Ginsberg et al. “Detecting influenza epidemics using search engine query data.” In: *Nature* 457.7232 (2009), pp. 1012–1014.

[44] Charles W Schmidt. *Trending now: using social media to predict and track disease outbreaks*. 2012.

[45] Shu-Chiung Hu et al. “A vehicular wireless sensor network for CO 2 monitoring.” In: *SENSORS, 2009 IEEE*. IEEE. 2009, pp. 1498–1501.

[46] You-Chiun Wang and Guan-Wei Chen. “Efficient data gathering and estimation for metropolitan air quality monitoring by using vehicular sensor networks.” In: *IEEE Transactions on Vehicular Technology* 66.8 (2017), pp. 7234–7248.

[47] Suhas Mathur et al. “Parknet: drive-by sensing of road-side parking statistics.” In: *Proceedings of the 8th international conference on Mobile systems, applications, and services*. 2010, pp. 123–136.

[48] Aravind Natarajan et al. “Measurement of Respiratory Rate using Wearable Devices and Applications to COVID-19 Detection.” In: *medRxiv* (2021).

[49] Yesin Sahraoui et al. “Performance evaluation of TCP and UDP based video streaming in vehicular ad-hoc networks.” In: *2018 International Conference on Smart Communications in Network Technologies (SaCoNeT)*. IEEE. 2018, pp. 67–72.

[50] Ali M Abdulshahed, Andrew P Longstaff, and Simon Fletcher. “The application of ANFIS prediction models for thermal error compensation on CNC machine tools.” In: *Applied Soft Computing* 27 (2015), pp. 158–168.

[51] Ali M Abdulshahed et al. “Thermal error modelling of machine tools based on ANFIS with fuzzy c-means clustering using a thermal imaging camera.” In: *Applied Mathematical Modelling* 39.7 (2015), pp. 1837–1852.

[52] Joel K Kelso, George J Milne, and Heath Kelly. “Simulation suggests that rapid activation of social distancing can arrest epidemic development due to a novel strain of influenza.” In: *BMC public health* 9.1 (2009), pp. 1–10.

[53] Jiaxu Zhang, Shaolin Hu, and Haoqiang Shi. “Deep learning based object distance measurement method for binocular stereo vision blind area.” In: *methods* 9.9 (2018).

[54] Abdelmoghit Zaarane et al. “Distance measurement system for autonomous vehicles using stereo camera.” In: *Array* 5 (2020), p. 100016.

[55] Deng-Yuan Huang et al. “Vehicle detection and inter-vehicle distance estimation using single-lens video camera on urban/suburb roads.” In: *Journal of Visual Communication and Image Representation* 46 (2017), pp. 250–259.

[56] Nobuo Sasaki, Syuhei Tomaru, and Shiori Nakamura. “Development of inter-vehicle distance measurement system using camera-equipped portable device.” In: *2017 17th International Conference on Control, Automation and Systems (ICCAS)*. IEEE. 2017, pp. 994–997.

[57] Jörg Wagner et al. “Multispectral Pedestrian Detection using Deep Fusion Convolutional Neural Networks.” In: *ESANN*. Vol. 587. 2016, pp. 509–514.

[58] Narinder Singh Punn et al. “Monitoring COVID-19 social distancing with person detection and tracking via fine-tuned YOLO v3 and Deepsort techniques.” In: *arXiv preprint arXiv:2005.01385* (2020).

[59] Muhammad Aqib et al. “A smart disaster management system for future cities using deep learning, GPUs, and in-memory computing.” In: *Smart Infrastructure and Applications*. Springer, 2020, pp. 159–184.

[60] Shaoqing Ren et al. “Faster r-cnn: Towards real-time object detection with region proposal networks.” In: *Advances in neural information processing systems* 28 (2015).

[61] Chaowei Wang et al. “Maximizing spatial–temporal coverage in mobile crowd-sensing based on public transports with predictable trajectory.” In: *International Journal of Distributed Sensor Networks* 14.8 (2018), p. 1550147718795351.

[62] Zhenqiang Xu et al. “TPSense: a framework for event-reports trustworthiness evaluation in privacy-preserving vehicular crowdsensing systems.” In: *Journal of Signal Processing Systems* 93.2 (2021), pp. 209–219.

[63] Lu Shao et al. “RTS: road topology-based scheme for traffic condition estimation via vehicular crowdsensing.” In: *Concurrency and Computation: Practice and Experience* 29.3 (2017), e3778.

[64] Ahmet M Elbir et al. “Vehicular Networks for Combating a Worldwide Pandemic: Preventing the Spread of COVID-19.” In: *arXiv preprint arXiv:2010.07602* (2020).

[65] Ezedin Barka et al. “Behavior-aware UAV-assisted crowd sensing technique for urban vehicular environments.” In: *2018 15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*. IEEE. 2018, pp. 1–7.

[66] Angelo Trotta et al. “Persistent crowd tracking using unmanned aerial vehicle swarms: A novel framework for energy and mobility management.” In: *IEEE Vehicular Technology Magazine* 15.2 (2020), pp. 96–103.

[67] Naser Hossein Motlagh, Miloud Bagaa, and Tarik Taleb. “UAV-based IoT platform: A crowd surveillance use case.” In: *IEEE Communications Magazine* 55.2 (2017), pp. 128–134.

[68] Minghu Zhang and Xin Li. “Drone-Enabled Internet of Things Relay for Environmental Monitoring in Remote Areas Without Public Networks.” In: *IEEE Internet of Things Journal* (2020).

[69] Rodolfo Vera-Amaro, Mario Eduardo Rivero-Ángeles, and Alberto Luviano-Juárez. “Data collection schemes for animal monitoring using WSNS-assisted by UAVS: Wsns-oriented or UAV-oriented.” In: *Sensors* 20.1 (2020), p. 262.

[70] Chunbo Luo et al. “A UAV-cloud system for disaster sensing applications.” In: *2015 IEEE 81st Vehicular Technology Conference (VTC Spring)*. IEEE. 2015, pp. 1–5.

[71] Adarsh Kumar et al. “A drone-based networked system and methods for combating coronavirus disease (COVID-19) pandemic.” In: *Future Generation Computer Systems* 115 (2020), pp. 1–19.

[72] Guanghao Sun et al. “Remote sensing of multiple vital signs using a CMOS camera-equipped infrared thermography system and its clinical application in rapidly screening patients with suspected infectious diseases.” In: *International Journal of Infectious Diseases* 55 (2017), pp. 113–117.

[73] Guanghao Sun et al. “Noncontact monitoring of vital signs with RGB and infrared camera and its application to screening of potential infection.” In: *Non-Invasive Diagnostic Methods-Image Processing* (2018).

[74] Chaochao Lu and Xiaoou Tang. “Surpassing human-level face verification performance on LFW with GaussianFace.” In: *Twenty-ninth AAAI conference on artificial intelligence*. 2015.

[75] Scott Morton, Ruben D’Sa, and Nikolaos Papanikolopoulos. “Solar powered UAV: Design and experiments.” In: *2015 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE. 2015, pp. 2460–2466.

[76] Sunghun Jung, Yonghyeon Jo, and Young-Joon Kim. “Flight time estimation for continuous surveillance missions using a multirotor UAV.” In: *Energies* 12.5 (2019), p. 867.

[77] Guglielmo Rossi et al. “Multitemporal UAV surveys for landslide mapping and characterization.” In: *Landslides* 15.5 (2018), pp. 1045–1052.

[78] 3GPP. *TS 23.285 V15.2.0. Technical Specification Group Services and System Aspects; Architecture enhancements for V2X services, Release 15*. Dec. 2018.

[79] Thirimachos Bourlai and Bojan Cukic. “Multi-spectral face recognition: Identification of people in difficult environments.” In: *2012 IEEE International Conference on Intelligence and Security Informatics*. IEEE. 2012, pp. 196–201.

[80] Thirimachos Bourlai et al. “Evaluating the efficiency of a night-time, middle-range infrared sensor for applications in human detection and recognition.” In: *Infrared Imaging Systems: Design, Analysis, Modeling, and Testing XXIII*. Vol. 8355. International Society for Optics and Photonics. 2012, 83551B.

[81] David A Berlin, Roy M Gulick, and Fernando J Martinez. “Severe Covid-19.” In: *New England Journal of Medicine* (2020).

[82] Xiaobing Wang et al. “Clinical characteristics of non-critically ill patients with novel coronavirus infection (COVID-19) in a Fangcang Hospital.” In: *Clinical Microbiology and Infection* (2020).

[83] Qiuhong Ke et al. “Computer vision for human–machine interaction.” In: *Computer Vision for Assistive Healthcare*. Elsevier, 2018, pp. 127–145.

[84] Mostafa Mehdipour Ghazi and Hazim Kemal Ekenel. “A comprehensive analysis of deep learning based representation for face recognition.” In: *arXiv preprint arXiv:1606.02894* (2016).

[85] Chaochao Lu and Xiaoou Tang. “Surpassing human-level face verification performance on LFW with GaussianFace.” In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 29. 1. 2015.

[86] Gary B Huang et al. “Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments.” In: ().

[87] *SUMO (Simulation of Urban MObility)*. `http://sumo.sourceforge.net`. accessed October 2, 2020.

[88] Dan Broyles, Abdul Jabbar, James PG Sterbenz, et al. "Design and analysis of a 3–D gauss-markov mobility model for highly-dynamic airborne networks." In: *Proceedings of the international telemetering conference (ITC), San Diego, CA.* 2010, pp. 25–28.

[89] P Jonathon Phillips et al. "The FERET database and evaluation procedure for face-recognition algorithms." In: *Image and vision computing* 16.5 (1998), pp. 295–306.

[90] Yi-Qing Wang. "An analysis of the Viola-Jones face detection algorithm." In: *Image Processing On Line* 4 (2014), pp. 128–148.

[91] Pavel Korshunov and Sébastien Marcel. "Deepfakes: a new threat to face recognition? assessment and detection." In: *arXiv preprint arXiv:1812.08685* (2018).

[92] KangGeon Kim et al. "Face and body association for video-based face recognition." In: *2018 IEEE Winter Conference on Applications of Computer Vision (WACV).* IEEE. 2018, pp. 39–48.

[93] Eunjoon Cho, Seth A Myers, and Jure Leskovec. "Friendship and mobility: user movement in location-based social networks." In: *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining.* 2011, pp. 1082–1090.

[94] Michael Y Li and James S Muldowney. "Global stability for the SEIR model in epidemiology." In: *Mathematical biosciences* 125.2 (1995), pp. 155–164.

[95] Hallam Stevens and Monamie Bhadra Haines. "Tracetogether: pandemic response, democracy, and technology." In: *East Asian Science, Technology and Society* 14.3 (2020), pp. 523–532.

[96] Anna U Morgan et al. "Remote Monitoring of Patients with Covid-19: Design, implementation, and outcomes of the first 3,000 patients in COVID Watch." In: *NEJM Catalyst Innovations in Care Delivery* 1.4 (2020).

[97] Rae Thomas et al. "Concerns and misconceptions about the Australian Government's COVIDSafe app: cross-sectional survey study." In: *JMIR public health and surveillance* 6.4 (2020), e23081.

[98] Yogesh Simmhan et al. "GoCoronaGo: privacy respecting contact tracing for COVID-19 management." In: *Journal of the Indian Institute of Science* (2020), pp. 1–24.

[99] Yesin Sahraoui et al. "A cooperative crowdsensing system based on flying and ground vehicles to control respiratory viral disease outbreaks." In: *Ad Hoc Networks* 124 (2022), p. 102699.

[100] Meghna Patel and Mitesh Patel. "An Integrated GIS/GPS Based e-Governance Approach for Different Agencies to Fight Against COVID-19." In: *Understanding COVID-19: The Role of Computational Intelligence.* Springer, 2022, pp. 447–469.

[101] Xueting Liao, Danyang Zheng, and Xiaojun Cao. "Coronavirus pandemic analysis through tripartite graph clustering in online social networks." In: *Big Data Mining and Analytics* 4.4 (2021), pp. 242–251.

[102] Rahul Goel and Rajesh Sharma. “Studying leaders & their concerns using online social media during the times of crisis-A COVID case study.” In: *Social Network Analysis and Mining* 11.1 (2021), pp. 1–12.

[103] Balsam Alkouz, Zaher Al Aghbari, and Jemal Hussien Abawajy. “Tweetluenza: Predicting flu trends from twitter data.” In: *Big Data Mining and Analytics* 2.4 (2019), pp. 273–287.

[104] Charlyn Villavicencio et al. “Twitter Sentiment Analysis towards COVID-19 Vaccines in the Philippines Using Naïve Bayes.” In: *Information* 12.5 (2021), p. 204.

[105] Kathrin Cresswell et al. “Understanding Public Perceptions of COVID-19 Contact Tracing Apps: Artificial Intelligence–Enabled Social Media Analysis.” In: *Journal of medical Internet research* 23.5 (2021), e26618.

[106] Enrique Hernández-Orallo et al. “Evaluating how smartphone contact tracing technology can reduce the spread of infectious diseases: the case of COVID-19.” In: *IEEE Access* 8 (2020), pp. 99083–99097.

[107] G Dumindu Samaraweera and J Morris Chang. “Security and privacy implications on database systems in big data era: A survey.” In: *IEEE Transactions on Knowledge and Data Engineering* 33.1 (2019), pp. 239–258.

[108] Med Abdelhamid Nedioui et al. “Detecting communities in social networks based on cliques.” In: *Physica A: Statistical Mechanics and its Applications* 551 (2020), p. 124100.

[109] Coen Bron and Joep Kerbosch. “Algorithm 457: finding all cliques of an undirected graph.” In: *Communications of the ACM* 16.9 (1973), pp. 575–577.