AI on the Move: From On-Device to On-Multi-Device

Huber Flores
University of Helsinki, Finland
huber.flores@helsinki.fi

Petteri Nurmi
Lancaster University, UK, and
University of Helsinki, Finland
p.nurmi@lancaster.ac.uk

Pan Hui
HKUST, Hong Kong
University of Helsinki, Finland
panhui@cse.ust.hk

***Abstract*—On-Device AI is an emerging paradigm that aims to make devices more intelligent, autonomous and proactive by equipping them with machine and deep learning routines for robust decision making and optimal execution in devices' operations. On-Device intelligence promises the possibility of computing huge amounts of data close to its source, e.g., sensor and multimedia data. By doing so, devices can complement their counterpart cloud services with more sophisticated functionality to provide better applications and services. However, increased computational capabilities of smart devices, wearables and IoT devices along with the emergence of services at the Edge of the network are driving the trend of migrating and distributing computation between devices. Indeed, devices can reduce the burden of executing resource intensive tasks via collaborations in the wild. While several work has shown the benefits of an opportunistic collaboration of a device with others, not much is known regarding how devices can be organized as a group as they move together. In this paper, we contribute by analyzing how dynamic group organization of devices can be utilized to distribute intelligence on the moving Edge. The key insight is that instead of On-Device solutions complementing with cloud, dynamic groups can be formed to complement each other in an On-Multi-Device manner. Thus, we highlight the challenges and opportunities from extending the scope of On-Device AI from an egocentric view to a collaborative, multi-device view.**

***Index Terms*—Cloud, Edge, Cloudlet, Artificial Intelligence, Device-to-Device, Data Analytics, Serverless**

I. Introduction

On-Device AI (Artificial Intelligence) is an emerging paradigm that aims to make devices more intelligent, autonomous and proactive by equipping them with machine and deep learning routines for robust decision making and optimal execution in devices' operations. On-Device intelligence promises the possibility of computing huge amounts of data close to its source, e.g., sensor and multimedia data. By doing so, devices can complement their counterpart cloud services with more sophisticated functionality to provide better applications and services[1,2]. Examples of this include, improved app management to extend battery life, enhanced connectivity and security, augmented natural interfaces for user interaction, more accurate sensing measurements from physical sensors and better personal assistant applications to support daily activities of users[3].

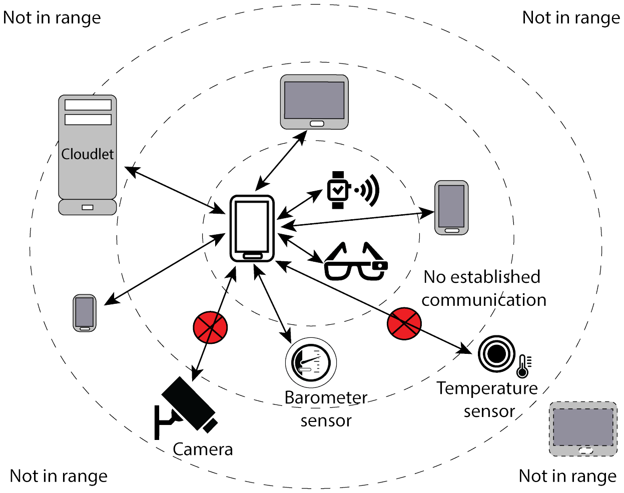

Fig. 1: Multi-device communications to engage into collaborative execution of tasks with heterogeneous devices. Issues of connectivity overhead, and heterogeneity of devices.

However, a fundamental limitation of On-Device solutions is that its execution solely depends on the isolated and scarce resources of a particular device, e.g., processing, storage, communication. Despite the computational power of smart devices being comparable to cloud servers [8], several work has demonstrated that AI routines like deep learning require heavy processing that drains the battery life of devices significantly [20]. Similarly, other work has also shown that while it is possible to reduce the cost of running AI in devices at some extent [19], devices still lack of enough resources to execute multiple AI routines on demand concurrently. Thus, it is very difficult to envision a proactive intelligence that covers different aspects of devices in a continuous manner. For instance, running at the same time, AI routines for object sensing monitoring, robust bio-metric user identification and extended remote control systems interactions.

On the other hand, the rapid proliferation of smart devices, e.g., smartphones, smart Tvs; and IoT devices, e.g., Raspberry

[1] https://www.technologyreview.com/hub/ubiquitous-on-device-ai/
[2] https://www.androidauthority.com/what-is-ai-893954

[3] https://www.qualcomm.com/news/onq/2018/08/08/edge-how-device-ai-enables-bright-biometrics-future

Pi[4], Arduino[5]; along with the emergence of services at the Edge of the network are driving the trend of migrating and distributing computation closer to end devices [30], [32], [35]. Indeed, devices can engage into opportunistic device-to-device (D2D) collaborations to reduce the burden of performing a common task by distributing its computation among the available devices. While this approach works at some extent, it relies on static and limited organization schemes to engage into collaboration [9], e.g., master/slaves [14], manager/workers [6]. As a result, when the number of nearby devices increases and it is too heterogeneous, interconnecting multiple devices to work together can become counterproductive rather than productive for resource optimization (Figure 1 illustrates the issue). Similarly, devices can also rely on Edge and Fog infrastructures, e.g., cloudlets [29], to process data instead of using remote cloud infrastructure. Since cloudlet infrastructure is hard to maintain and deploy, it does not provide a dense support like the ubiquitous one provided by the cloud. Thus, cloudlet solutions are not well adopted yet into existing architectures. As the matter of fact, cloudlets are encountered by devices in the wild in the same way as any other device, the only difference is that when a device finds Edge support via cloudlet, it offloads the highest possible processing to the cloudlet instead of splitting the task among the available devices (including itself) [7], [11].

Certainly, D2D collaborations in multi-device environments are critical to reduce energy footprint of applications and execute resource intensive tasks without cloud support. However, most of D2D collaborations are opportunistic (temporal basis) and its benefits depend highly on the right self-organization of devices. While several work has shown the benefits of opportunistic encounters of a device with others [3], [18], not much is known regarding how devices can be organized as a group as they move together. In this paper, we contribute by analyzing how dynamic group organization of devices can be utilized to distribute intelligence in the wild. The key insight is that instead of On-Device solutions complementing with cloud, dynamic groups can be formed to complement each other in an On-Multi-Device manner. Moreover, On-Multi-Device intelligence can be also envisioned to assign intelligence to groups, such that interaction between groups allow devices to execute different types of AI routines concurrently. We believe that by relying on collaborative multi-device approaches for running AI, it is possible to create a distribute intelligence on the Edge that is more scalable and less resource intensive for devices.

II. Background & Related Work

Mobile, wearable and IoT devices rely on their constrained resources to perform isolated computational operations in the wild. To overcome these limitations and provide better services, end devices connect to cloud to request functionality in the form of services (see Figure 2a). While the cloud provides the platform for ubiquitous access of services, cloud infrastructure is typically located several or many network hops away from devices. As a result, the cloud is encountering issues for meeting the Quality-of-Service (QoS) and Quality-of-Experience (QoE) requirements. For example, the response time requirements of many systems and applications, e.g., health, transport, and autonomous cars, cannot be fulfilled by services with unpredictable or significant latency. End devices in turn are experiencing resource degradation in terms of e.g., battery and performance, for accessing remote cloud services. To address these problems, infrastructure at the Edge of the network is being investigated. As shown in Figure 2a, the main idea is that cloud functionality can be migrated from the cloud to the Edge, such that devices can access services at low latency, which reduces operational costs of devices and improves response time of applications.

On the other hand, increased computational capabilities of smart devices along with improved frameworks for running AI routines, it is opening a new plethora of On-Device solutions with more intelligent applications and services that just complement partly with the cloud/Edge in a serverless manner [12] (see Figure 2b). This enable devices the possibility to become more autonomous and proactive to users' needs. However, given the resource limitations of devices and the complexity of deploying ubiquitous Edge infrastructure, it is difficult to envision a large diversity of AI services for devices. Thus, in this paper, we explore the foundational research challenges and opportunities of using groups of multiple devices to distribute a large scale variety of intelligent services on the move and on the Edge. By doing this, a service can be host by different groups that complement each other, such that the service can exist completely on the Edge in an ubiquitous manner to devices (see Figure 2c). We review existing work in these lines as follows.

On-Device AI: Several work has study the feasibility of running deep learning on mobile, wearable and IoT devices [19], [20], [23], [28]. Other work also has explored the possibility of running AI routines using opportunistic collaboration of devices [18]. Since end devices are resource constrained, several work has proposed instead to run AI on Edge/Fog [30]. On-Device solutions have been proposed mostly in industry to improve the provisioning of services and applications running in mobile devices. The main idea is to run learning frameworks, such as TensorFlow [1] in devices, and complement functionality with cloud computing. In contrast to other work, we explore the large-scale distribution of AI routines in the wild and on the Edge by harnessing multi-device groups.

Proximal and collaborative infrastructures: Infrastructure in the form of a cloudlet [33] has been envisioned for supporting computation of devices on the Edge, e.g., cyber foraging [2]. Since edge cloudlets are hard to deploy and maintain densely, frameworks for remote offloading have been explored [4], [7], [13], [17]. More recently however, to avoid oscillating network latency, D2D offloading has been encouraged [9], [14]. Social-aware systems rely on opportunistic

[4] https://www.raspberrypi.org/

[5] https://www.arduino.cc/

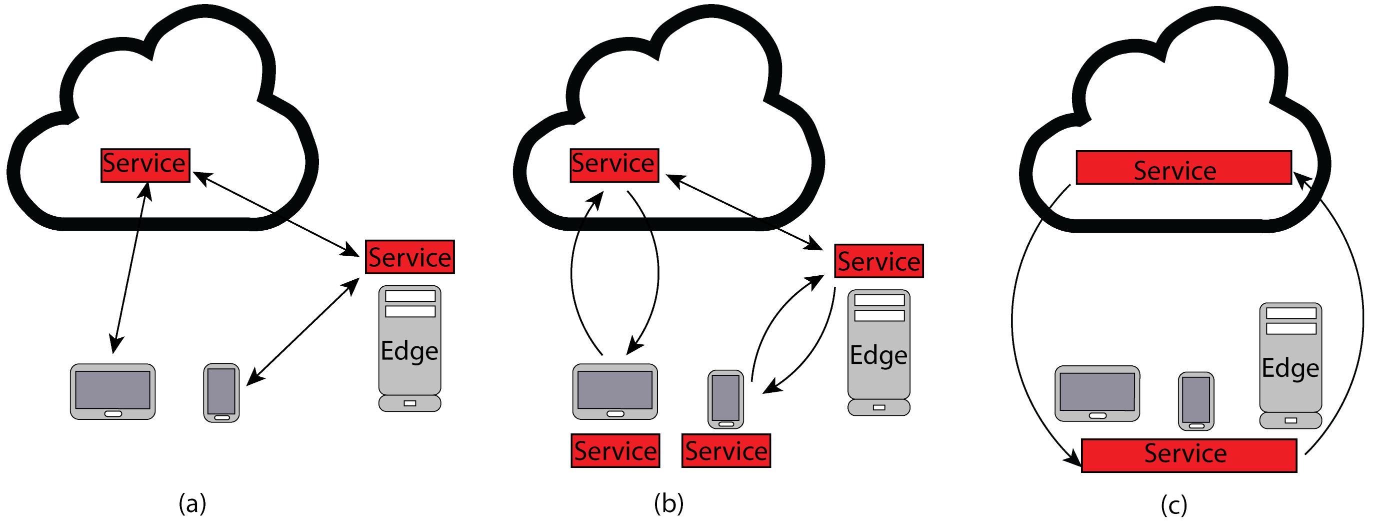

Fig. 2: Device service schemes, a) Device dependent, service is located solely in the cloud; b) On-Device, service is distributed in the cloud and the device, both complement each other; c) On-Multi-Device, service is located in the cloud and distributed among devices, both are independent on their own, and just synchronize to keep consistency between them.

D2D communications to build a decentralized self-organizing network based on opportunities of social proximity and interactivity [9]. Commonly these kind of systems are mostly investigated for recovery and disaster scenarios [24]. The main goal of these systems is providing a dynamic network infrastructure in which each device can perform network functions by themselves [27], e.g., packet forwarding, routing, and distributed storage. Other examples of such systems include Mobile adhoc networks (MANET) [25], wireless sensor networks [34], Massive-multiplayer online games (MMOG) [15], Peer-to-peer (P2P) systems [26], Delay-tolerant networks [22], and Decentralized Online Social networks (DOSN) [5]. More recently, however, these systems have been investigated for harnessing the computational resources of devices in conjunction to create a computational infrastructure on the edge that is in proximity of users (HAGGLE [31], HyMobi [9], FemtoClouds [14], Ubispark [18]). The main problem of these systems is fault tolerance for proximal interactions, which just have been explored partially for networking [3] and computation [9]. Unlike previous work, in this paper, we focus on identifying how devices move together as groups, such that groups can be used to distribute intelligence on the Edge. We explore the challenges and opportunities of extending the scope of On-Device from a single to a multi-device view.

III. From On-Device to On-Multi-Device: Challenges and Opportunities

Cloud computing is a fundamental platform for the provisioning of ubiquitous services over the Internet. However, over the years, end devices are becoming less dependent on the cloud to perform complex computations. Indeed, as the amount of data increases, processing data locally becomes less costly for devices rather than moving it completely to a remote location. Thus, devices are adopting approaches that lead to less interactions with the cloud, e.g., Serverless AWS Lambda[6].

On the other hand, the cloud is migrating part of its functionality to the Edge. However, since an Edge infrastructure is far from being ubiquitous for devices due to its deployment complexity, its adoption within existing architectures is arguable. Alternatively, collaboration between devices in the wild has the potential to address most of the problems about performing complex computations on the Edge. Several studies have shown that smart devices are frequently co-located in proximity to at least one other device throughout the day, suggesting that devices can potentially collaborate to reduce the effort of resource intensive tasks, e.g., sensing [21], offloading [10], networking [16], storage, etc. However, distributing a task over a group of devices to be provided as a service, it is a very complex matter. As devices have different mobility patterns, it is necessary to identify devices that share a tight mobility relation. In other words, group of devices that host a service need to be formed by devices that move together for long periods. In addition, once a group of devices is identified, then devices need to be assessed based on several factors, e.g., computational power, sensing quality, communication technologies, etc., to determine the amount of resources that will contribute with the hosting of the service, and whether resources from different devices can work together without degrading service provisioning. In this section, we highlight these challenges and highlight potential solutions.

A. Identifying duration in proximity patterns for multi-device infrastructure formation

The formation of a multi-device infrastructure depend on the proximity of devices to engage into collaborations (see

[6] https://aws.amazon.com/lambda/

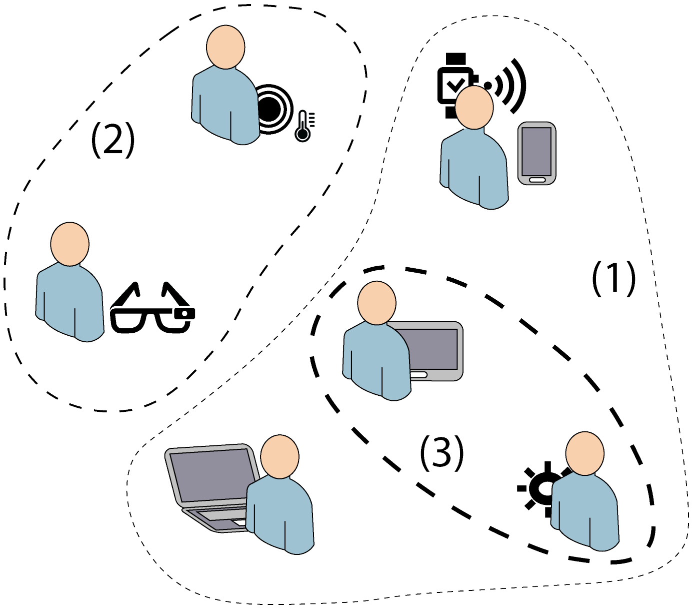

Fig. 3: Identification of proximity patterns to form multi-device infrastructures. We can observe that three different groups can be identified. Each group can then host a particular service that executes AI functionality. Groups interact between them to consume and scale services.

Figure 3). As an example, personal devices are likely to remain close to each other throughout most of the day, whereas proximity of other devices depends on the social relationship between the owners of the devices. Other devices are encountered based on the daily routine mobility patterns of users, e.g., static devices like IoT and cloudlets. Proximal and temporal duration of devices is critical in a collaboration as the processing of a task needs to be finished before devices are out of reach of each other. Otherwise, applications can become resource intensive as they can waste additional energy to recover or simply applications become unusable due to execution crashes. Since devices have different mobility patterns, it is important to identify different levels of duration in the interaction between devices. Deploying a service over a multi-device infrastructure requires then devices to have a tight mobility between them. Otherwise, a service is unable to be executed due to missing devices.

B. Addressing devices in a multi-device interaction

One critical problem of collaborative D2D systems, it is addressing devices in proximity. Usually, to engage into a D2D collaboration, a device need to first detect devices in proximity, so that then it can proceed to establish communication channels with every device available. While this approach is reasonable to merge resources of different devices, in practice, a device can address a limited amount of devices before the collaboration process becomes too resource intensive. When this occurs, a collaboration between devices become counterproductive rather than productive. By identifying devices moving in groups a priori, it is possible to mitigate the problems of addressing devices, such that it is possible to obtain a more stable yet distributed infrastructure.

C. Splitting a service over a multi-device infrastructure

Existing solutions for D2D collaboration rely on splitting a task equally between devices for its execution (Figure 4a). However, splitting a service over a multi-device infrastructure requires different considerations. Indeed, once a group of devices with tight mobility patterns is identified, then it is possible to quantify the amount of resources to be contributed by each device. Since devices can differ between each other depending on the capabilities of their resources, it is possible to encounter problems of resource fragmentation. This can impede sharing of resources between devices. Resource fragmentation can occur if resources of devices are too different between each other. For instance, the processing capabilities of a smartwatch are lower when compared with a smartphone. Thus, if both devices engage into collaborative processing, then the smartwatch will become a bottleneck in the system. As a result, each device needs to be assigned with a part of the service that is in proportional with its capabilities as shown in Figure 4b. However, assigning a part that is proportional to the device capabilities, it is a difficult issue as it involves to take into consideration many parameters of devices, specially during runtime, e.g., memory available, CPU and transmission rate, among others.

D. Opportunistic distribution of AI over multi-device groups

By identifying different groups of devices, it is possible to assign each group to provision a particular AI functionality. Interactions between groups then can be used to exchange AI services between devices. By doing this, it is possible to create scalable infrastructures in which a variety of AI is available on the Edge. In addition, since a particular device can belong to multiple groups, it is possible to assign a particular AI functionality based on its group characteristics (see Figure 5). For instance, a device can be the most powerful device within one group, but the less resourceful in another one. Similarly, a device that belongs to a group during night could have more available resources, e.g., processing time, bandwidth, when compared with a device that belongs to a group during the day.

On the other hand, a service distributed in parts over a multi-device infrastructure introduces several opportunities to develop and design a more scalable and energy-efficient system to distributed intelligence that is available on the Edge. A multi-device infrastructure can be used to disseminate reusable parts of a service to other groups that have similar multi-device configurations. For instance, Figure 6 shows a device migrating a part of a service from a cloudlet to another cloudlet, which is in a different location, such that other devices can complement each other with that functionality.

The utilization of reusable parts of a service that can be used to complement others, it is a powerful approach to execute resource intensive functionality on the Edge. By doing so, devices avoid issues of assigning roles, e.g., master/slaves, and

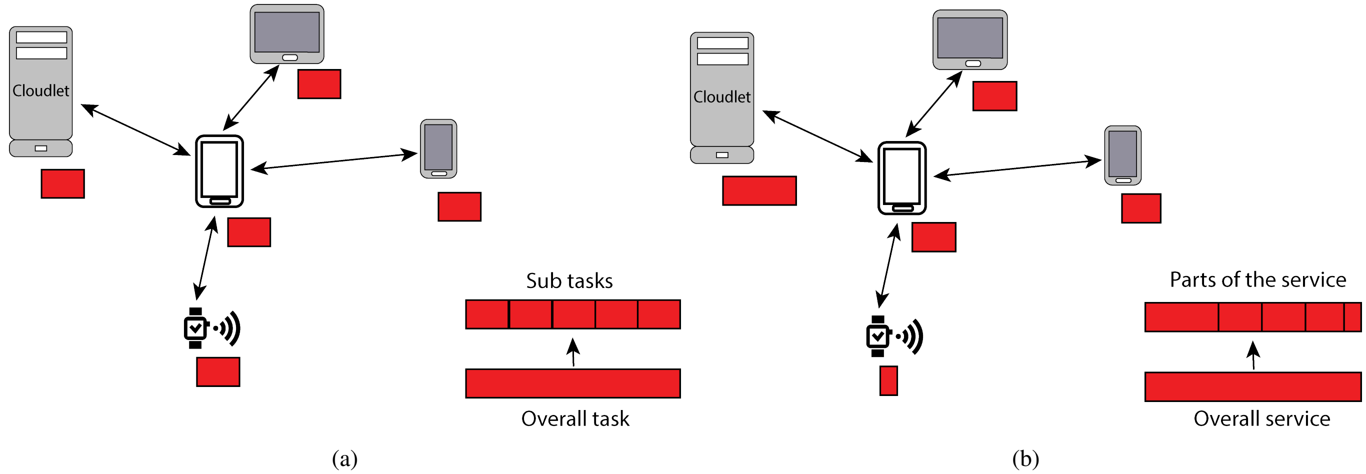

Fig. 4: Distribution of computation among multiple devices in the wild. (a) Division of a traditional computational task in a multi-device environment, (b) Optimal distribution of a task as a service over a multi-device infrastructure, where a device host part of the service based on its computational capacity.

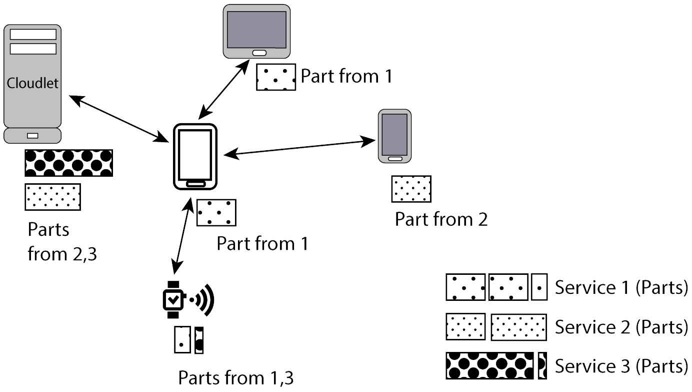

Fig. 5: Assigning the provisioning of a particular AI service to a multi-device group. From the multi-device infrastructure above, three different services can be hosted in multiple devices. A particular device can host more that one service (part) when belonging to different groups at the same time.

calculating the optimal execution of a task as in traditional collaborative systems, instead, devices are just concern on executing their assigned part based on its group.

IV. Conclusions

In this paper, we contribute by exploring the challenges and opportunities from extending the scope of On-Device intelligence, from a single to a collaborative On-Multi-Device perspective. While On-Device intelligence can be achieved in a single device by complementing part of its functionality with remote cloud infrastructure, devices can still experience degradation in performance due to oscillating changes in communication with remote services. While devices can rely on Edge and Fog infrastructure as well as co-located collaborative devices to create a computational infrastructure closer to end devices, existing solutions can be more

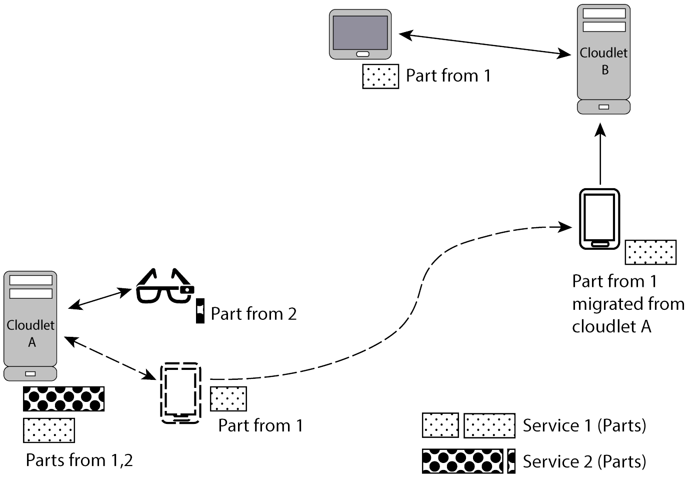

Fig. 6: Opportunistic migration of a service part from one group to another. By taking advantage of users mobility, reusable functionality can be then disseminated by devices themselves.

counterproductive rather than productive for devices as they lack of proper mechanisms to scale in the wild and provide ubiquitous support. In contrast to current work, we envision that by identifying group of devices that move together, it is possible to distribute intelligence in these group of devices, such that devices complement each other instead of using remote infrastructure. This opens a new plethora of large scale distributed intelligence that is available on the Edge for devices.

Acknowledgment

This work was partially funded by the Academy of Finland grant 317875, and by the Jorma Ollila Grant. This article only reflects the authors views.

References

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al. Tensorflow: a system for large-scale machine learning. In *OSDI*, volume 16, pages 265–283, 2016.

[2] Rajesh Balan, Jason Flinn, Mahadev Satyanarayanan, Shafeeq Sinnamohideen, and Hen-I Yang. The case for cyber foraging. In *Proceedings of the 10th workshop on ACM SIGOPS European workshop*, pages 87–92. ACM, 2002.

[3] Marco Conti, Silvia Giordano, Martin May, and Andrea Passarella. From opportunistic networks to opportunistic computing. *IEEE Communications Magazine*, 48(9), 2010.

[4] Eduardo Cuervo, Aruna Balasubramanian, Dae-ki Cho, Alec Wolman, Stefan Saroiu, Ranveer Chandra, and Paramvir Bahl. Maui: making smartphones last longer with code offload. In *Proceedings of the ACM International Conference on Mobile systems, applications, and services (MobiSys 2010)*, San Francisco, CA, USA, June 15–18, 2010.

[5] Anwitaman Datta, Sonja Buchegger, Le-Hung Vu, Thorsten Strufe, and Krzysztof Rzadca. Decentralized online social networks. In *Handbook of Social Network Technologies and Applications*, pages 349–378. Springer, 2010.

[6] Niroshinie Fernando, Seng W Loke, and Wenny Rahayu. Honeybee: A programming framework for mobile crowd computing. In *International Conference on Mobile and Ubiquitous Systems: Computing, Networking, and Services*, pages 224–236. Springer, 2012.

[7] Huber Flores, Pan Hui, Petteri Nurmi, Eemil Lagerspetz, Sasu Tarkoma, Jukka Manner, Vassilis Kostakos, Yong Li, and Xiang Su. Evidence-aware Mobile Computational Offloading. *IEEE Transactions on Mobile Computing*, 2017.

[8] Huber Flores, Pan Hui, Sasu Tarkoma, Yong Li, Satish Srirama, and Rajkumar Buyya. Mobile code offloading: from concept to practice and beyond. *IEEE Communications Magazine*, 53(3):80–88, 2015.

[9] Huber Flores, Rajesh Sharma, Denzil Ferreira, Vassilis Kostakos, Jukka Manner, Sasu Tarkoma, Pan Hui, and Yong Li. Social-aware hybrid mobile offloading. *Pervasive and Mobile Computing*, 36:25–43, 2017.

[10] Huber Flores, Rajesh Sharma, Denzil Ferreira, Chu Luo, Vassilis Kostakos, Sasu Tarkoma, Pan Hui, and Yong Li. Social-aware device-to-device communication: a contribution for edge and fog computing? In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct*, pages 1466–1471. ACM, 2016.

[11] Huber Flores and Satish Srirama. Adaptive code offloading for mobile cloud applications: Exploiting fuzzy sets and evidence-based learning. In *Proceeding of the fourth ACM workshop on Mobile cloud computing and services*, pages 9–16. ACM, 2013.

[12] Huber Flores, Xiang Su, Vassilis Kostakos, Jukka Riekki, Eemil Lagerspetz, Sasu Tarkoma, Pan Hui, Yong Li, and Jukka Manner. Modeling mobile code acceleration in the cloud. In *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on*, pages 480–491. IEEE, 2017.

[13] Mark S Gordon, D Anoushe Jamshidi, Scott Mahlke, Z Morley Mao, and Xu Chen. Comet: code offload by migrating execution transparently. In *Proceedings of the USENIX Conference on Operating Systems Design and Implementation (OSDI 2010)*, Hollywood, CA, USA, October 08–10, 2012.

[14] Karim Habak, Mostafa Ammar, Khaled A Harras, and Ellen Zegura. Femto clouds: Leveraging mobile devices to provide cloud service at the edge. In *2015 IEEE 8th International Conference on Cloud Computing (CLOUD)*, pages 9–16. IEEE, 2015.

[15] Thorsten Hampel, Thomas Bopp, and Robert Hinn. A peer-to-peer architecture for massive multiplayer online games. In *Proceedings of 5th ACM SIGCOMM workshop on Network and system support for games*, page 48. ACM, 2006.

[16] Bo Han, Pan Hui, and Aravind Srinivasan. Mobile data offloading in metropolitan area networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 14(4):28–30, 2011.

[17] Sokol Kosta, Andrius Aucinas, Pan Hui, Richard Mortier, and Xinwen Zhang. Thinkair: Dynamic resource allocation and parallel execution in the cloud for mobile code offloading. In *Proceedings of the Annual IEEE International Conference on Computer Communications (INFOCOM 2012)*, Orlando, Florida, USA, March 25–30, 2012.

[18] Eemil Lagerspetz, Huber Flores, Niko Mäkitalo, Pan Hui, Petteri Nurmi, Sasu Tarkoma, Andrea Passarella, Jörg Ott, Peter Reichl, Marco Conti, et al. Pervasive communities in the internet of people. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 40–45. IEEE, 2018.

[19] Nicholas D Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, Lei Jiao, Lorena Qendro, and Fahim Kawsar. Deepx: A software accelerator for low-power deep learning inference on mobile devices. In *Proceedings of the 15th International Conference on Information Processing in Sensor Networks*, page 23. IEEE Press, 2016.

[20] Nicholas D Lane, Sourav Bhattacharya, Petko Georgiev, Claudio Forlivesi, and Fahim Kawsar. An early resource characterization of deep learning on wearables, smartphones and internet-of-things devices. In *Proceedings of the 2015 international workshop on internet of things towards applications*, pages 7–12. ACM, 2015.

[21] Youngki Lee, Younghyun Ju, Chulhong Min, Seungwoo Kang, Inseok Hwang, and Junehwa Song. Comon: Cooperative ambience monitoring platform with continuity and benefit awareness. In *Proceedings of the ACM International Conference on Mobile Systems, Applications, and Services (MobiSys 2012)*, (Low Wood Bay, Lake District, United Kingdom), June 25–29 2012.

[22] Yong Li, Guolong Su, Dapeng Oliver Wu, Depeng Jin, Li Su, and Lieguang Zeng. The impact of node selfishness on multicasting in delay tolerant networks. *IEEE Transactions on Vehicular Technology*, 60(5):2224–2238, 2011.

[23] Titti Malmivirta, Jonathan Hamberg, Eemil Lagerspetz, Xin Lin, Ella Peltonen, Huber Flores, and Petteri Nurmi. Hot or not? robust and accurate continuous thermal imaging on flir cameras. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom 2019)*, Kyoto, Japan, March 11–15, 2019.

[24] Abraham MartíN-Campillo, Jon Crowcroft, Eiko Yoneki, and Ramon Martí. Evaluating opportunistic networks in disaster scenarios. *Journal of Network and Computer Applications*, 36(2):870–880, 2013.

[25] Pietro Michiardi and Refik Molva. Core: a collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. In *Advanced communications and multimedia security*, pages 107–121. Springer, 2002.

[26] Thomas Moscibroda, Stefan Schmid, and Rogert Wattenhofer. On the topologies formed by selfish peers. In *Proceedings of the twenty-fifth annual ACM symposium on Principles of distributed computing*, pages 133–142. ACM, 2006.

[27] Luciana Pelusi, Andrea Passarella, and Marco Conti. Opportunistic networking: data forwarding in disconnected mobile ad hoc networks. *IEEE communications Magazine*, 44(11), 2006.

[28] Daniele Ravi, Charence Wong, Benny Lo, and Guang-Zhong Yang. A deep learning approach to on-node sensor data analytics for mobile or wearable devices. *IEEE journal of biomedical and health informatics*, 21(1):56–64, 2017.

[29] Mahadev Satyanarayanan, Victor Bahl, Ramón Caceres, and Nigel Davies. The case for vm-based cloudlets in mobile computing. *IEEE pervasive Computing*, 2009.

[30] Mahadev Satyanarayanan, Pieter Simoens, Yu Xiao, Padmanabhan Pillai, Zhuo Chen, Kiryong Ha, Wenlu Hu, and Brandon Amos. Edge analytics in the internet of things. *IEEE Pervasive Computing*, (2):24–31, 2015.

[31] James Scott, Jon Crowcroft, Pan Hui, and Christophe Diot. Haggle: A networking architecture designed around mobile users. In *WONS 2006: Third Annual Conference on Wireless On-demand Network Systems and Services*, pages 78–86, 2006.

[32] Xiang Su, Pingjiang Li, Jukka Riekki, Xiaoli Liu, Jussi Kiljander, Juha-Pekka Soininen, Christian Prehofer, Huber Flores, and Yuhong Li. Distribution of semantic reasoning on the edge of internet of things. In *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom)*, pages 1–9. IEEE, 2018.

[33] Tim Verbelen, Pieter Simoens, Filip De Turck, and Bart Dhoedt. Cloudlets: Bringing the cloud to the mobile user. In *Proceedings of the third ACM workshop on Mobile cloud computing and services*, pages 29–36. ACM, 2012.

[34] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer networks*, 52(12):2292–2330, 2008.

[35] Wenxiao ZHANG, Huber Flores, and HUI Pan. Towards collaborative multi-device computing. In *2018 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 22–27. IEEE, 2018.