International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 6 (2019) pp. 1220-1243

An Extensive Review on Sensing as a Service Paradigm in IoT: Architecture, Research Challenges, Lessons Learned and Future Directions

Sampath Kumar Y R[1] and Champa H N[2]

Abstract

The Internet of Things is regarded as a trending technology of the modern age, through which several sensors are connected to the Internet for productive and compelling asset administration in a Smart Environment (viz., Smart Home, Smart city etc.). Programming applications, infrastructures and platforms are rendered as services through Cloud technologies. Sensing as a Service (SaS) is an advanced type of distributed computing that avails the worldwide resources of IoT and it facilitates creation of a shared network of sensors that are available as a consumable service. These services are leveraged by the organizations and developers, thereby users are provided with an opportunity to monetize the data using existing infrastructure. Moreover, there's a need to investigate the idea of SaS paradigm and analyze how it suits with the IoT technology. The main focus is to explore the utility of SaS model in innovative, social, technological and ideological points of view. In this extensive survey, we analyze different paradigms of sensing frameworks and provide a holistic view on various challenges and issues that incur in designing an ideal framework so that all the requirements of the stakeholders (users, developers, operators and organization) are met in best possible ways.

Keywords:- Cloud-centric IoT, Cloud of Things, Crowdsensing, Selective Sensing, Sensor selection formulation.

1. INTRODUCTION

Sensing as a Service (SaS) is an IoT-based cloud service prototype that facilitates access to the IoT. It can also promote an efficient delivery of sensor data based on demand via a data stream to different stake holders to meet the requirements. The IoT comprises of virtually interconnected objects or smart devices that are remarkably physically identifiable and are fit for sensing, processing and interacting with the events. An application that is based on IoT domain (ecosystem) can be developed by integrating the services of smart devices [1]. Sensors in Smart devices are used to perform tasks of an IoT application. These sensors accumulate data around their deployment environment. Data collectors or sensors obtain necessary information through an IoT application on a pay-as-you-go fashion and the data distribution and collection are managed via cloud. The real time data gathered from sensor devices is monetised and transformed into new data streams so that the accuracy of predictive services are enhanced and the network operations are also optimized to facilitate effective automation. The essential segments of the sensing model require efficient search methodologies, and effective sensing algorithms for providing the sensing service.

Sensing is an element of IoT and it can be described as the process of collecting data from smart devices within a network[2]. In the present era of IoT, everything is connected to each other and APIs assist users to link and integrate services. Analyzing, managing, utilizing and capturing the data remain as most pressing hurdles of the IoT. There is a need to structure and architect IoT so that these systems can have many points of interest from the virtually unlimited capacities which the Cloud Computing can offer, for instance, to repay the constraints pertaining to technology (viz., storage, processing, energy consumption) of the smart devices. IoT and cloud have numerous complementary qualities emerging from the distinctive proposals in the literature. The Cloud has the potential to function as a transitional layer between the entities of IoT and the applications in order to manage the resources effectively.

1.1 Motivation

The convergence between IoT and Cloud Computing reveals a new idea that appears to be promisingly named in the literature as *IoT-cloud*. The number of smart devices associated with Internet is growing exponentially since recent years, and as of now it has surpassed the number of people on the Earth. According to [3], the count of smart devices has already reached 12 billion and it is speculated to grow at a faster rate in couple of years. With the emergence of number of IoT systems, different viewpoints reveal several inspirations on its integration between IoT and Cloud Computing infrastructures. Preserving information locally and temporarily won't be possible any longer and there will be a requirement for huge storage space. In addition to this, huge measure of information can't be handled locally in the devices as their computational capacities are limited. The functionalities of Cloud-enabled technologies can be leveraged by integrating the IoT services within the cloud. Also, the data gathered by the smart devices is huge and deploying these devices into infrastructure requires a huge cost, both in terms of capital investments and time to setup. Thus, it is impossible for every IoT application to have its own infrastructure of smart devices to accomplish its needs. Cloud computing technology can be advantageous, as different IoT applications share a common infrastructure and utilize only those part of the infrastructure (*i.e* smart devices) that they require to meet the demand. To this end, SaS paves the way to facilitate the various demands of stakeholders by offering them to use the IoT infrastructure (devices/services) based on their requirement. Having been motivated by this model of Service provisioning, we discuss few recent works related to this model and thus identify the present needs in form of problems and challenges that are to be resolved in future, so that an efficient and effective service provisioning model is designed for the IoT.

1.2 Contributions

The significant **contributions** with respect to this paper are delineated as follows:

- Fundamentals of SaS model: The conceptual layers of SaS are explained along with its functionalities.
- Challenges pertaining to SaS model: The various challenges which are necessary to be addressed in the process of designing a service model are discussed.
- Review of present literature on SaS: Different existing Frameworks for sensing along with middlewares that assist in the governance of SaS are discussed and compared. Various ideas for integrating IoT technology in sensor devices are extensively reviewed and key technologies with respect to SaS (like crowdsensing and mobile sensing) are discussed in brief.

1.3 Organization

The remaining part of this paper is systematized as follows. Section 2 deals with the generic architecture of SaS model. Section 3 describes the different scenarios and ideologies of sensing frameworks. Section 4 discusses various challenges pertaining to SaS environment. Whereas Section 5 deals with insights gained from the comparative study with respect to different sensing frameworks, middlewares for sensing, integration of Cloud and IoT, Crowdsensing techniques and Mobile sensing technologies. Whereas Section 6 contains the concluding remarks and it also explains the relevance of SaS with respect to IoT.

2. SaS MODEL

Sensing as a service is a modern concept which deals with providing sensing services using sensors and other physical devices that are connected to IoT. It can be visualised to a marketplace where sensor data is traded with interested consumers in a transparent manner. SaS can also be compared to a platform through which data owners can exchange or vend data to interested sensor data customers in a 'pay as you-go' fashion. The platform facilitates development of sensor deployments and shares the common IoT infrastructure so that the data is stored, processed and collected efficiently. Due to the sharing of IoT infrastructure, it decreases the cost of acquiring sensor data. The fundamental components of SaS architecture are the cloud platform and worker nodes. The overall sensing tasks are controlled by the platform whereas the worker nodes execute the sensing tasks as per the requirements of cloud IoT platform.

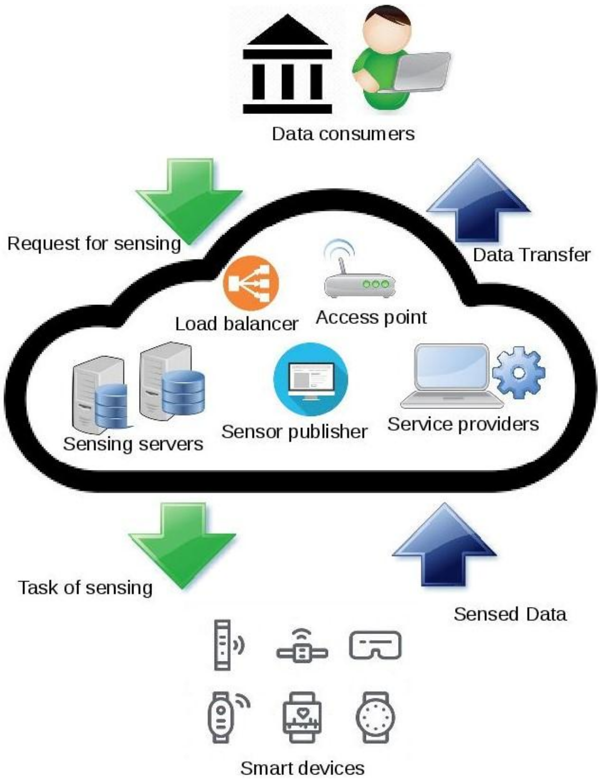

Figure 1: SaS Model

The generic architecture of a sensing model is represented in figure 1. The deployment of multiple sensing servers enables to manage the sensing requests from various geographical locations. A sensing request is initiated by a cloud user through a web based application from either a computer or sensor device, later it is transmitted to the required destination. Data that is gathered from IoT devices are stored in cloud and it is transferred to the required users. A significant aspect of this feature is that the sensing tasks are on par with the sensing requests from other sensor users.

2.1 Architectural Requirements of the SaS Model

1. The model should provide a layered system that's capable of handling the environmental factors and network channel through optimal power consumption.
2. Essential abstractions must be provided to carryout the fundamental operations across a wide range of heterogeneous sensor devices.
3. A consistent means to integrate higher level systems in order to exchange sensor information should be supplied.
4. An ideal model should have the potential to manage both dynamic and static networks.

2.2 Functionalities supported by SaS model

1. Web Interface: Collecting and requesting the essential information from cloud users is necessary so that it is

effortlessly accessible via computer or mobile devices.

2. Tracking mobile devices: There is a need to maintain information such as number of available sensors, locations, residual energy and list of mobile devices that are indulged in sensing task. A suitable interface designed between mobile devices and sensing servers facilitates to gather the sensed data and further it also enables to push sensing tasks to mobile devices.
3. Generating Sensing Tasks: Based on the request of users, advanced sensing functions are generated.
4. Recruiting mobile users: A group of mobile users are recruited to participate in sensing process when there is a demand for new sensing schedule.
5. Scheduling Sensing Activities: A scheduling algorithm or policy is used to schedule the sensing activities of each sensing task that is associated with a set of smart devices.
6. Managing Sensors: Deployment of an application on each sensor device enables the operation of its sensors in order to perform the various requested sensing actions such as collection of sensed data and transmitting it to the sensing server.
7. Storing and handling the data: The sensed data must be obtained from sensor devices and useful information is stored in database.

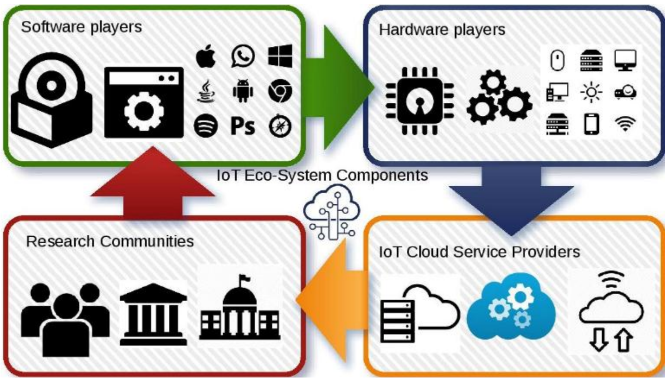

Figure 2: Business model for SaS

Figure 2 explains the business model of SaS that includes important components like hardware, software, research communities and IoT-cloud service providers.

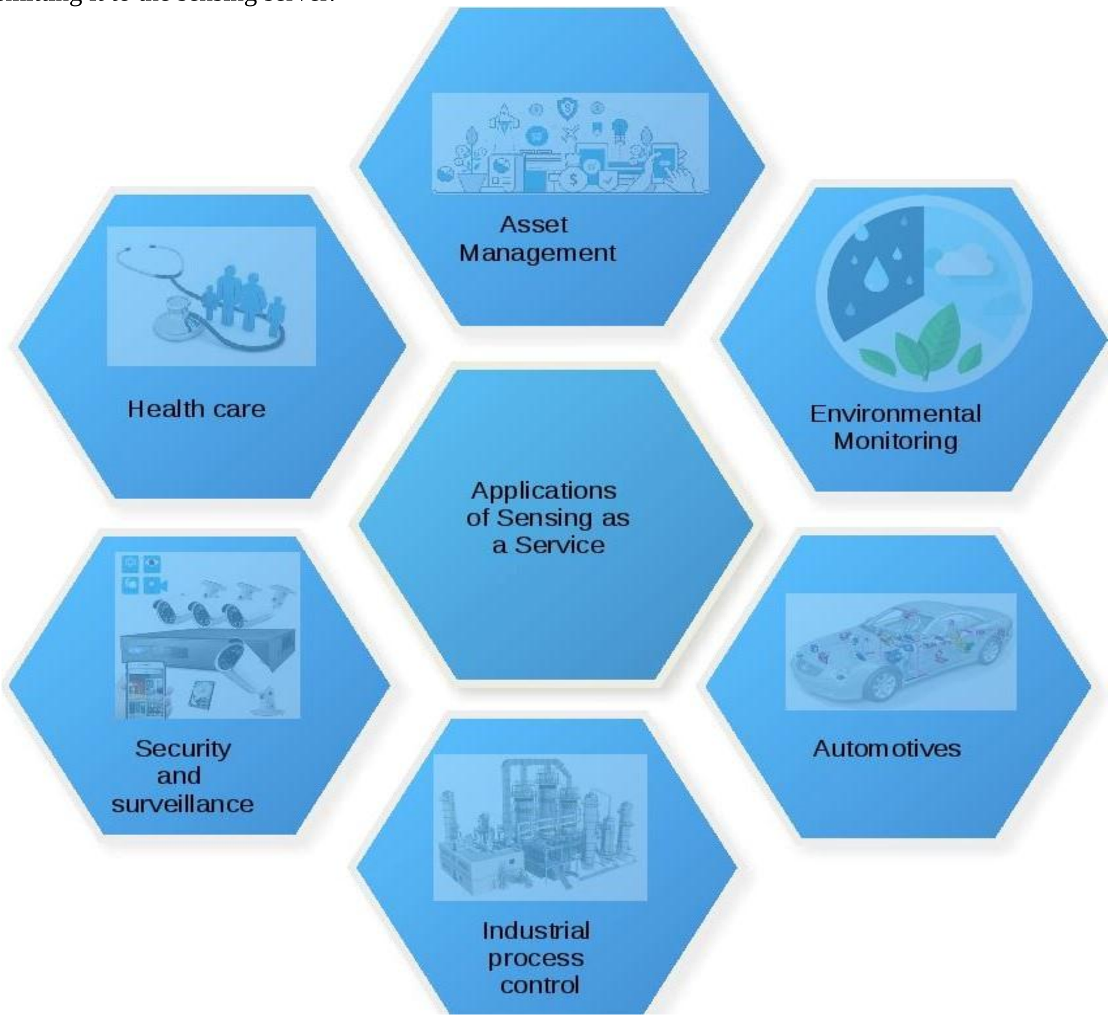

Figure 3: Applications of SaS in modern age

Figure 3 highlights the different scenarios where the model of SaS is utilized.

2.3 Benefits and advantages of the sensing model

1. Integrated cloud computing: The sensing model has the potential to inherit the functionalities of significant cloud models like IaaS, PaaS and SaaS.
2. Participatory Sensing and actuation: Sensors are deployed rapidly across vast geographical locations so that the workload is effectively distributed and the task of capturing different phenomena is simplified.
3. Reusing and sharing: Sensing model inhibits the characteristic features that stimulate the concept of sensor data distribution and sharing.
4. Data monetization: Existing network assets are utilized to design new data streams, thus real time data pertaining to an environment is monetized.

3. REVIEW AND ANALYSIS OF COMPONENTS AND DIFFERENT PERSPECTIVES IN SAS

In this particular module, we present a systematic review of the recent research publications that are associated with SaS model from different perspectives of sensing framework, cloud middleware, IoT and cloud integration, crowd sensing and mobile sensing along with their applications. For each work we have identified its concept, contributions and potential pitfalls.

3.1 Paradigms of Sensing Framework

An ideal framework for sensing supports features like remote configuration, data uploading, fast implementation, encryption for potentially sensitive data and basic survey system for manual data collection. Internet of Things framework supports the interaction between devices in a distributed computing environment.

Mafrur *et al.,* [4] suggested a sensing framework that enables an user and the application developers to interact and share the information, however, data collector used in this methodology is restricted to twenty probes and addition of further probes requires more resources like hardware, power and processors. The proposed system fails to analyze and visualize the data and also it fails to address the issues like user's privacy and computation time. An entity that utilizes climate overlay approach to combine cloud infrastructure and wireless sensor networks was presented by Arunachalam *et al.,*[5][6][7] according to which the sensing service framework is controlled by climatic data and WSN components were deployed by employing virtual sensors. Positioning of the cloud users inside the data center enables the protection of user data and virtualization of network sensors helps to avoid the deployment of redundant WSN, however, integrating cloud and WSN reduces the recovery cost and only independent components and databases that are connected to cloud are at high risk.

Hassan *et al.,* [8] suggested a technique to connect people, sensors and software in order to develop sensing applications that require integration of sensor networks and cloud or data center, nevertheless, a pub-sub pattern is proposed in order to simplify the process of unifying WSN with the cloud dependent applications. To match the published events with subscriptions, an effective and ascendable group indexing technique is preferred, however, the data pertaining to real world applications cannot be utilized using this approach. Xiang Sheng and Jian Tang [9] manifested a model that provides various sensing services by utilizing smart devices. The proposed system demonstrated how sensing can be offered as a service and it was portable on different hardware platforms, however, scripting language was used to describe the sensing tasks.

Chattopadhyay *et al.,* [10] implemented an algorithm for collecting, aggregating and assembling the contextual data but the network based rule engines used in that process suffers from scalability issue and thus it's not suitable to be deployed on a large scale. Niyato *et al.,* [11] introduced a mechanism to harvest energy based on IoT sensing service. The suggested methodology increases the hit rate pertaining to a sensing function, however, the energy consumption is reduced since the caching mechanism decreases the total number of requests that are consigned to the sensor. A cloud architecture that is capable of virtualizing various categories of sensing infrastructures was popularized by Fazio *et al.,* [12] to offer advanced services to assist Homeland Security cloud. Although the proposed methodology is feasible and scalable, it does not have a prototype that is capable of filtering the information according to user's profile.

Gubbi *et al.,* [13] presented his vision to implement internet of things by utilizing cloud computing technology and they also demonstrated how convergence in several interdisciplinary technologies are necessary to implement IoT. In this approach, the consumers can select the type of service by modifying the standards of service parameters. The user centric framework helps in differentiating the networking, storage and visualization layers, however, this method fails to work in a shared environment. Chifor *et al.,* [14] introduced a platform using social network reputation analysis according to which an end point is able to publish a message that contains sensed data and a witness message. The suggested solution can be parameterized and it is scalable, nevertheless, the testing phase lacks a proper scripting solution. Sang *et al.,* [15] developed an algorithm by considering both the sensor and user context, however, due to the vast number of available points of interest and user intent, the identification of local recommendations is tedious and cumbersome.

A framework proposed by Zhang *et al.,* [16] contains energy and budget constraints that helps in detecting coarse and fine-grained event boundaries. The technique adopted here is based on probability, thus the contribution of participants can be estimated accurately and the proposed algorithm for event detection can be improved in distributed manner to attain a solution that ensures quality of sensory data and reduce the user disturbance, however, the participants are selected by considering factors like quality of interest, mobility patterns and user incentive requirements. A distributed framework to enable domain mapping of given access requests was proposed by Ghosh *et al.,* [18] and it provides a feasible solution to the inter-domain role mapping issue. The proposed technique has efficient response time, and it also provides scalable cloud based services, however, the mapping of required permissions from a distant user to local users is effectively handled by the proposed method and the role set is generated in polynomial time.

Table 1. Comparison of Different Sensing Frameworks

Authors	Research Concepts	Benefits	Limitations
Mafrur *et al.* [4]	Framework based on opportunistic method for collecting data	Framework functions like a bridge between user and application developers or researchers	Data collector is limited to 20 probes, and problems such as data management, user's privacy, information processing and data visualization are not handled
Arunachalam *et al.* [5]	Framework for sensing that is deployed through utilization of the virtual sensors.	Low recovery cost, climate updates are available and user data is protected	Cloud components and independent databases are at jeopardy
Zhang *et al.* [16]	Sensing framework which adopts two-step heuristic solution have energy and budget constraints	Contribution of participants is estimated through probability based methods	Quality of sensory data cannot be ensured till the algorithm is improved
Wan *et al.* [17]	Cloud services rely on inter-cloud architecture, and an IoT sensing framework exhibits scalability at computation level	Anomalous events are detected efficiently and effective communication is ensured to users during emergency	Risk factors are evaluated and efficiency is improved by utilizing the probabilistic theory.
Ghosh *et al.* [18]	Access requests are mapped via collaborating domain in a distributed environment	Requested permissions from a remote user to local systems are mapped and handled by the framework	Role set in the framework is generated faster *i.e.* in polynomial time.

Table 1 denotes the comparison between the different sensing frameworks. A framework that is dependent on architectural model of IoT for sensing heterogeneous data was proposed by Al Fagih *et al.*,[19]. Here, the pricing utility function is used for acquiring data by considering the resource limitation factor in terms of capacity, delay and existence of data, however, the data is traded based on quality and delay requirements, and also the proposed method maximizes service provider's profit and optimizes the network resources.

Tsai *et al.*, [20] built a platform to utilize the sensory data from physical entities or objects and humans for predicting the occurrence of disasters. Here, the interaction between participants and systems are managed by the system or crowd, however, the integration of sensor data from both the humans and physical sensors is not supported in the suggested platform. A sensor network abstraction layer was presented by Philipp *et al.*, [21] to create versatile public sensing systems for executing random queries. By utilising a virtual sensor concept, sensor network can be redesigned to support different sensing systems. However, location uncertainity increases the overall average distance of all techniques. Hsu *et al.*, [22] investigated on a context-aware sensor selection method which is based on credentials of sensor attributes and the mechanism of publish services. According to this approach the essential sensors were selected among huge set of accessible sensors based on various sensor selection and search methods. The suggested architecture helps in intelligent selection of relevant sensors for users depending on the requirement or necessity of users and it also facilitates to acquire longer network lifetime, compared to that of traditional keyword based search techniques and the users can determine certain conditions in their requests, however, machine learning techniques are not included in the suggested method.

Sim *et al.*, [23] devised a technique to distinguish between sufficient and adaptive concession rate. The suggested method is efficient as it has higher advantage than the traditional mechanisms, however, to evaluate the performance of this technique, an agent-based Cloud testbed is utilized along with the Java language. A negotiation strategy to support cost and time-slot negotiation among cloud agents was presented by Son *et al.*, [24]. Here, Agents are designed to make proposals and aggregated utility is generated in negotiation round. Empirical results illustrate that price and time slot negotiation entities come to quick agreements and achieve higher utilities than other techniques. Wei *et al.*, [25] showed how service-oriented computing intents to utilise services as essential blocks to build low-cost but riskless and dependable rapid applications in a cloud computing environment. Organisations and companies can develop software systems by capturing the essential services dynamically but the combination of cloud computing and service oriented computing challenges the ideologies about enterprise computing. A multicloud environment explained by Le *et al.*, [26] deals with the mechanisms for configuring and automating activities but it fails to optimise the selection of cloud resources at runtime.

A technique for patient-centric healthcare services that is proposed by Zhang *et al.*,[27] relies on big data and cloud technologies. The system comprises of data management layer, a data collection layer, an interface for users and an API for developers. A community-oriented platform designed by Zhang *et al.*,[28] supports sensor data as a service by utilizing Web 2.0 platform and the users are permitted to reuse data and integrate it. The suggested platform consists of data model to support interoperable and scalable sensor data, however the metadata related to platform and sensor data are disassociated and stored separately. A tool to configure and discover sensors deployed in particular location was proposed by Perera *et al.*,[29] and it supports the interaction between hardware and cloud-based IoT middleware. The connectivity between software system and sensor hardware is established by configuring the sensors. Factors like sampling rate, number of sensors, dynamicity, context and data acquisition help to optimise sensing and data communication, however, processes such as Sensor discovery and sensor configuration are automatically monitored by utilising key factors like count of sensors, number of on demand schedules, heterogeneity, dynamicity and data acquisition methods. A model which relies on the security certification scheme for the cloud was proposed by Anisetti *et al.*, [30] and this scheme supports autonomic cloud computing systems that can be self-repaired, self-monitored, self-optimised and their non-functional properties can be tuned, nevertheless, there's a need for techniques that possess the potential to control the trust information.

3.2 Middleware for SaS

The middleware can be described as the software that exists between applications and operating systems such that the complexity of network environment is hidden and also the application does not handle memory management, protocol handling, parallelism and network functionality. A middleware for IoT must be generic in nature, adaptable, reflective and scalable. Cloud of Things can be referred to a middleware that describes the different attributes of Internet of Things by exploring the synergy between IoT and cloud computing. There exists several platforms like Linked Sensor Middleware(LSM) that combines the sensor data with the semantic web in an unified model, however, the data access and sensor search is done using a web interface and SPARQL. Global Sensor Networks(GSN) is one platform which facilitates integration of heterogeneous sensing technologies and abstraction of the data collection process by utilizing virtual sensors which are offered as a service. The selection of sensors is possible through a web interface whereas the sensor is identified and discovered using keywords. Platforms such as OpenIoT uses GSN in order to access the data using different ontologies and also to search the data using various dynamic methods.

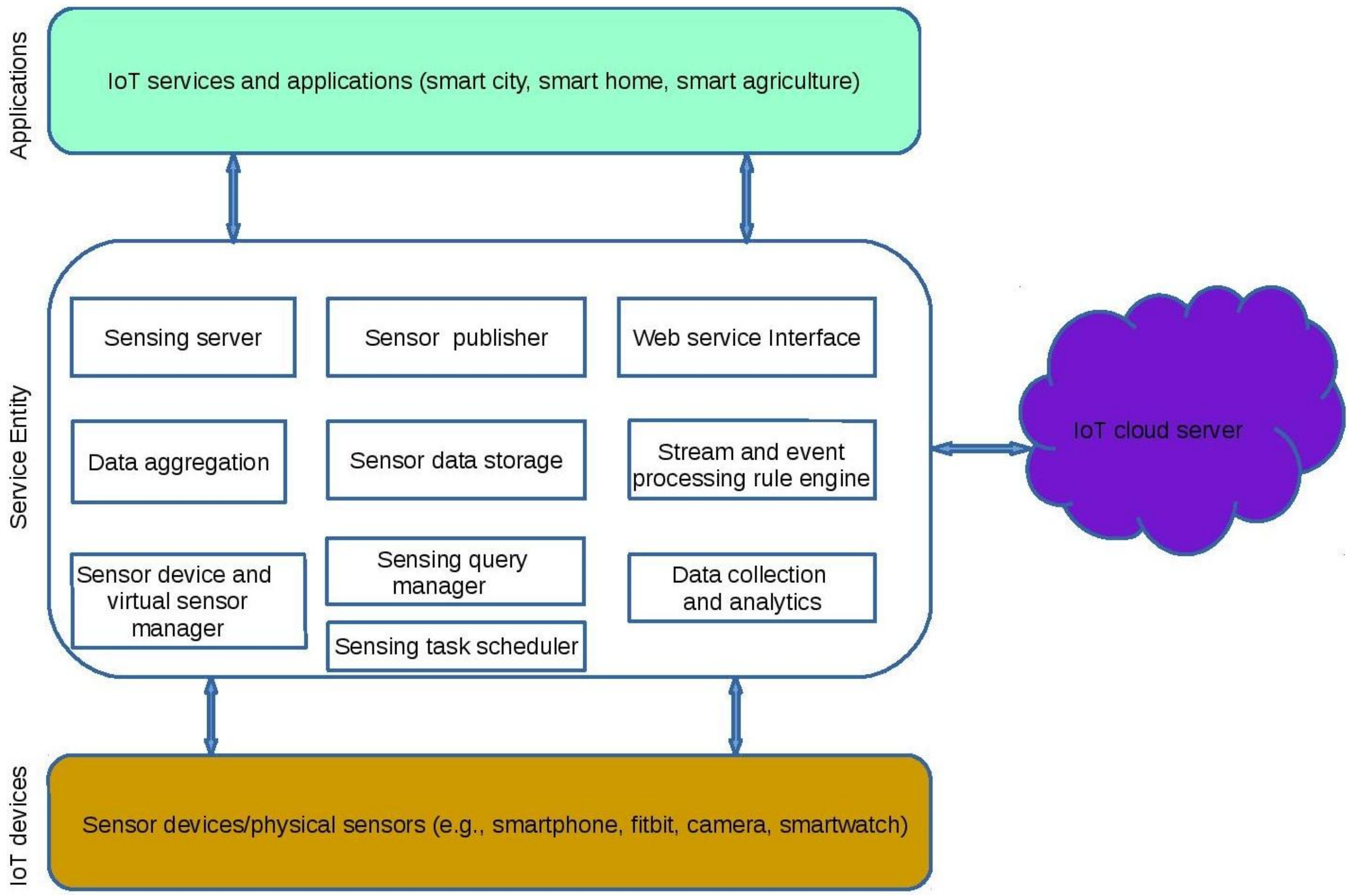

Figure 4: Sensing based service oriented middleware for SaS

Figure 4 shows an architecture of a sensing based middleware which comprises of different entities like physical sensors, IoT device managers, sensing task scheduler, sensing query manager, stream processing engine, sensing server, sensor publisher, web service interface, IoT applications and IoT cloud server. It also includes several phases such as data aggregation, data collection and data storage.

Table 2. Comparison of Different Cloud of Things Middleware

Authors	**Research Concepts**	**Benefits**	**Disadvantages**
Abdelwahab *et al.* [31]	Memory models from the IoT devices were derived from LTE-optimized replication protocol.	Potential to reduce the energy utilisation, delay and price of cloud-offloading.	Energy utilisation and delay are increased at memory level.
Kim *et al.* [32]	Middleware that can logically isolate and share multi-tenant applications via virtualization.	Overhead of virtualization is minimized due to an efficient access control and sharing mechanisms	Multi-tenant applications depend on different contexts of execution.
Hu *et al.* [33]	Low-priced participatory sensing system which uses an aggregation of portable mobile sensor units, mobile apps and cloud computing.	Entity can improve user engagement in exposure management.	Currently it is deployed in an environment with low density and it fails to capture sporadic pollution events.
Patti *et al.* [34]	Platform for controlling and maintaining resource constraints in public spaces	Supports an easy integration of heterogeneous technologies	Middleware designed is generic and event based
Braubach *et al.* [35]	An integrated model for programming is used for enhancing the non functional properties of cloud based applications	Addresses the solutions for monitoring and engineering the non functional properties	System does not support automation

Table 2 depicts the comparision between various Cloud of Thing’s Middlewares. Abdelwahab *et al.,* [31] introduced a cloud architecture and a protocol which can manage the overheads of LTE by introducing a delay and utilizing the memory efficiently by reducing the amount of energy consumed and cost of cloud offloading, however, the proposed idea relies on the technology of communication between the devices and it is also based on the ideology of memory replication in the IoT devices. A middleware that is capable of virtualizing physical smart devices for handling dependency between smart devices was designed by Kim *et al.,* [32] to isolate multiple applications. The proposed middleware monitors the smart devices by resolving resource conflicts among them and it also supplies a model for efficient utilization of the shared smart devices, however, virtualization is achieved through the efficient sharing and logical isolation of several multi-tenant applications, and the smart devices are efficiently utilised depending upon the different contexts of multi-tenant applications.

Hu *et al.,* [33] evaluated and described the design of a system which collectively integrates mobile sensor units, cloud computing, smart-phones and mobile applications in order to model, calculate and access the information related to air pollution. The proposed model utilizes mobile applications and web-based tools for estimating and visualizing the exposure of air pollution on individuals. This type of system can expand user engagement in management of vulnerabilities, however, one of the limitations is that currently it cannot be deployed and also it is not capable of capturing infrequent pollution events..

Braubach *et al.,*[35] presented an approach for enhancing cloud related applications, however, the method and model of an autonomic system is not handled by a global scheduler and the Quality of Service is calculated by the components and application. Burak *et al.,* [36] described that the sensing services which are provided through IoT objects are based on a cloud platform that delivers sensing data as resources. The proposed model helps to search sensors by estimating prospective localities of the service providers depending on their network interaction, however, the methodology to search and recruit sensing service providers are not addressed in this approach. Fazio *et al.,* [37] designed a cloud dependent architecture which is capable of processing the data and virtualizing different types of sensing infrastructures. The suggested middleware offers services for storage, computation and sensing and it also provides relevant information about data by reducing the tariffs of processing, communication and storage in the cloud, however, this middleware does not support data aggregation and filtering. Zhou *et al.,* [38] proposed a Cloud-dependent platform which can accommodate other cloud delivery models for managing, accelerating, and developing an IoT application in order to facilitate an user to access and control things or IoT objects from anywhere. The proposed model provides the ability to deploy, run, develop and manage applications of several devices that are interconnected via the Cloud and it also supports to integrate technologies like Cloud and IoT, in order to provide a feasible approach to build an application.

Lea *et al.,* [39] introduced the general idea of IoT hubs, where a group of multiple hubs are connected to build systems which connect a variety of smart devices for bidirectional communication between IoT application and the devices. Here, an IoT framework is introduced as an hub and the main emphasis was given to build a middleware which could be utilized in Smart City research, however, there is a need to ensure that the proposed smart city hub has the proficiency to adopt several services and the main problems which need to be resolved are support for hybrid public cloud and interoperability. Pereira *et al.,* [40] presented an integrated network architecture that consists of heterogeneous Embedded Internet Systems which include sensor nodes that are transmitted via bluetooth and can communicate using standard protocols. The actuator and sensor data are exchanged by using an internet-based cloud Service architecture and user's local cloud, which consists of smart phones, laptops and IoT sensor devices. The proposed architecture enables the transmission of sensor data to SOA in real time, however, the idea of true mobility is accomplished by the deployment of user's smartphone and mobile access network.

Rao *et al.,* [41] described the methodology to integrate cloud and IoT to overcome the challenges in Big Data. SaS on cloud was illustrated by utilizing certain applications like smart agriculture, Augmented Reality and Environment monitoring. An architecture model was designed to offer SaS on cloud platform and the Cloud is defined as an authentic, favourable and productive solution to track, connect and manage the IoT. Sensor cloud combines WSN with cloud computing infrastructures using sensing applications, however, some of the challenges of sensor cloud are scaling the real-time data, handling Large scale computing framework, hardware upgradation and Complex Event processing. The disadvantage of scientific cloud for IoT is that it's not possible to potentially increase and decrease the capacity of a particular server. Taivalsaari *et al.,* [42] introduced an end-to-end architecture in which the sensors provide data about the entity that are detected. The architecture comprises of actuators which can be defined as the constituents that alter the position of a physical object through energy consumption and it also consists of Hubs or Gateways which can be referred as the devices that are used for pre-processing, assembling and fetching data from surrounding IoT devices and their sensors. IoT systems basically depend on connectivity, however, due to the absence of universal interoperable software development environments, a single IoT application written by a developer is not portable on all types of devices.

Zhu *et al.,* [43] recommended a mechanism to share resources in cloud by utilizing an algorithm to perform independent and periodic tasks in the clouds. Here, a cloud platform was used for conducting experiments and the performance of proposed technique was analysed using workloads from previous versions, however, the task scheduling problem in real time is solved efficiently using virtualized clouds and the time required for communication and dispatching are not considered in addition with integration of scaling down policy for improving the resource utilization. Zhang *et al.,* [44] recommended a middleware to realize data fusion of mobile agents on the nodes of WSNs. Although the utilization of mobile agents in WSNs has several advantages like reliability, task-adaptability, progressive accuracy and scalability, the mobile agents can only be operated on physical sensor nodes but cannot be simulated on a system. Fargo *et al.,* [45] suggested an autonomic power and performance management technique to dynamically match applications for cloud systems. The proposed approach resulted in significant power reduction and the utilization of static workload is high, however, frequency scaling method decreases the amount of power consumed. Aazam *et al.,* [46]recommended a model of Cloud of Things which aims to manage all constraints like advanced reservation, resource prediction refunding, allocation and pricing, however, the model was implemented and assessed using CloudSim and the systematic resource management becomes more critical. Laukkarinen *et al.,* [47] proposed the features for actuating and measuring IoT devices and predicted that an embedded cloud was a solution in order to measure, distribute and expand resources of different IoT technologies, however, the proposed design is not scalable on different platforms.

3.3 IoT-Sensor and Cloud Integration

IoTCloud is a platform which can interact with data available in real time through data analysis frameworks that are employed in cloud to facilitate real time data processing. This platform is also capable of managing the device and data stream. Smart devices are connected to cloud services to process and control the data. An user can develop real time processing algorithms, without the knowledge of how the data is distributed and transferred, as the sensor cloud allows users to utilize sensors without worrying about the other details. Sensor cloud is an entity that focuses on managing sensors via cloud by providing sensor management capabilities such as checking the usage of physical sensors. It also facilitates the sensor data to process, store and categorize efficiently such that it is cost-effective, available based on necessity, and easily accessed by the stakeholders.

An architecture that's capable of capturing sensor data by utilizing smartphone sensors was proposed and manifested by Perera *et al.,* [48]. It consists of a middleware that operates on android platform for gathering the sensor data, however, it is generic and the raw sensor data can be altered and queried along with the remaining sensors that are connected to it but it's installation and configuration require definite processing power. A lightweight mobile application was proposed by Perera *et al.,* [49] to connect and retrieve sensor data from external sensors by utilising wireless communication techniques like bluetooth and WiFi. The sensor data from a mobile application is combined, controlled and processed by the GSN server, thus there is a reduction in significant amount of manual labour as the system is configured automatically, however, in this approach the IoT middleware is unaware of the details of sensors that are connected to mobile phones.

Doukas *et al.,* [50] developed and presented a platform to collect data and store it on an open cloud infrastructure such that it is further used for monitoring and processing. The suggested solution is cost-effective and simple but securing the private data and maintaining the energy constraints of the micro controller are the problems that need to be addressed or resolved. An approach to increase the productivity of IoT services with sensor cloud infrastructure, was suggested by Yuriyama *et al.,* [51]. Here, the new sensor services are created by service requests, however, the layers of cloud infrastructure may alter the service time and functionality of the model. The service distribution in an IoT-cloud network is modeled as minimum cost mixed-cost flow problem by Barcelo *et al.,* [52] and the resources of IoT-cloud network were also characterised depending on their associated sensing and computing attributes. High reliability, reduced operational cost, low latency, high flexibility, support for mobility, scalability and location awareness are the main advantages of IoT-cloud networks, nevertheless, collecting global information about demands of the service by a centralised network controller and dispersing the result to all nodes of network is a overhead.

Nastic *et al.,* [53] preferred a technique that defines abstractions for effective growth of IoT applications and it also aims to provide an effective solution for provisioning of different IoT related applications on a cloud. Cloud assisted IoT applications can be implemented by developers without worrying about the difficulty of managing raw data streams and device services, however, it lacks suitable code distribution mechanism or procedure. A cloud-integrated sensor networking architecture was suggested by Phan *et al.,* [54] in order to virtualize sensors in cloud and to operate physical sensors. Here, the various layers of cloud, sensor devices and edge communicate by utilizing push-pull hybrid approach, however, the multi-objective analysis technique cannot fabricate and control such an hybrid communication.

A mixed approach to balance success rate and adequacy, suggested by Zheng *et al.,* [55] shows a certain level of intelligence to attain a high performance, however, this method is subjected to more failures if the data is inefficient and incomplete. Wang *et al.,* [56] demonstrated how cloud computing and IoT can assist a conventional system in order to transform it into an advanced system. The proposed modularized architecture makes the system robust, reliable, flexible and automated.

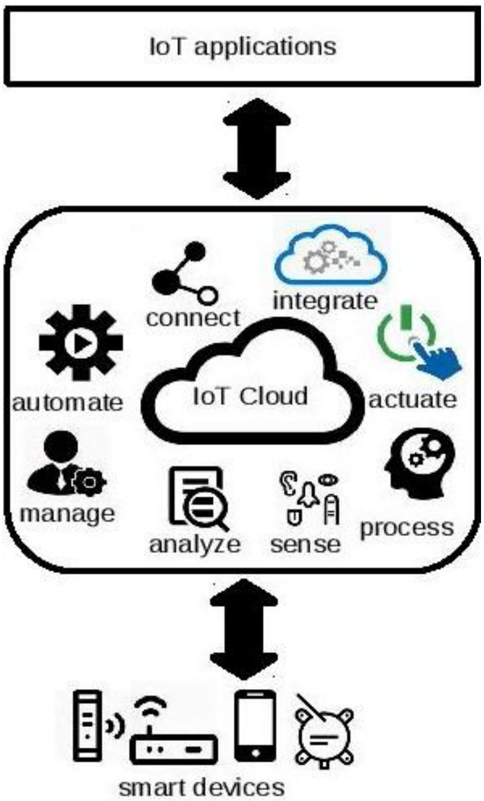

Figure 5: Integration of IoT services in Cloud

Figure 5 illustrates how the sensor data collected from IoT devices are integrated in a cloud platform so that it can be utilized by an IoT application. Truong *et al.,*[57] explored how IoT and cloud systems provided a suitable layer for execution, deployment and provisioning of applications for different domains. Cloud services provide computational data processing and management platform for IoT. The IoT providers differ from cloud service providers with respect to software layers, communication protocols, and provisioning models as the sensors and their corresponding services belong to different providers and are hosted in different repositories, however, the software fails to deal with multilevel software stack configuration.

A publish-subscribe model was drafted by Roffia *et al.,* [58] to support interoperability in smart applications. Here the subscribers and publishers use standard SPARQL queries which are built on a generic SPARQL architecture, however, the architecture is natively parallel and it also supports the non-occurrence of an event within a stipulated time.

A cloud based solution which considers link quality and spatiotemporal correlation of data for integrating different data was suggested by Bijarbooneh *et al.,* [59] for reducing the energy incurred for collecting data. The performance of protocol is assessed in terms of data utility, accuracy of data prediction and energy efficiency. Data is collected at non-uniform frequencies or random intervals in Belief Propagation based approach. Path quality and correlation are balanced to restrict the data utility devices under dynamically increasing loads, such that the edge can be configured to specific deployment for supporting dynamic load variations, however, the local interface of both application and data can be obtained by a cloud and also the sensor system and interaction system can be analysed and monitored efficiently.

An architecture was introduced by Xu *et al.,* [60] to scale IoT under dynamically increasing loads, such that the edge can be configured to specific deployment for supporting dynamic load variations, however, the local interface of both application and data can be obtained by a cloud and also the sensor system and interaction system can be analysed and monitored efficiently.

A plugin based IoT middleware that is built on android platform was proposed by Perera *et al.,* [61] for collecting and processing the sensor data and it is also capable of working on resource constraint smart devices. The data collected by various sensor devices is analyzed and processed before it is transmitted over a network, however, it is difficult to extend and interoperate between the software systems as they remain a closed source and the devices cannot interoperate with solutions provided by different vendors.

Table 3. Comparison of Different Techniques of Sensor Cloud integration

Authors	**Research Concepts**	**Benefits**	**Limitations**
Roffia *et al.* [58]	Architecture built on a generic SPARQL to support information level interoperability across smart IoT applications	Architecture is natively parallel and facilitates event negation	Maximum event load affordability relies on the number of available cores and the potential to reject unrelated events
Bijarbooneh *et al.* [59]	Cloud based solution for integration of different data is performed by considering the entities like spatio-temporal interaction of data and link quality.	High data quality is ensured by ASBP protocol as it decreases energy consumption by activating only few nodes in the network	Strategy is based on heuristics and the greedy technique cannot maintain a route for all nodes in the network
Xu *et al.* [60]	Scaling behavior of the cloud architecture is demonstrated under dynamic conditions by considering various IoT applications.	Architecture is scalable across different IoT use case scenarios.	Cloud-sensor system is capable of obtaining a local view of both the applications and data.
Perera *et al.* [61]	IoT middleware for mobile devices that is capable of acquiring and processing the sensor data periodically.	Can be operated on mobile devices for the purpose of gathering and processing the sensor data.	Interoperability and extensibility of software systems is not possible as they remain a closed source.
Vinh *et al.* [62]	Applications of sensors, mobile devices and Cloud computing are supported via service architecture by integrating IoT and Mobile Cloud Computing technologies.	Service architecture is generic and thus supports a variety of Smart City based IoT applications, and it also guarantees the scalability and security for certain scenarios	Higher security requirements cannot be satisfied for all scenarios.

Table 3 depicts the comparision between different strategies for sensor cloud integration.

Vinh *et al.*, [62] introduced service oriented architecture to integrate mobile cloud computing and IoT such that it supports the applications of sensors, cloud computing and mobile devices. The proposed work focusses on developing a service architecture that offers enhanced services to users and services for assuring security and scalability in smart city applications, however, this approach does not satisfy higher security requirements when compared with other schemes. A caching algorithm suggested by Xu *et al.*, [63] selectively caches application fragment from cloud to edge to enhance cloud scalability in cloud sensors but it fails to address the challenges like sensor energy constraint and cloud scalability. A technique to integrate IoT in enterprise services, which was presented by Spiess *et al.*, [64] is event based and the implemented architecture conceals the heterogeneity of software, hardware and communication protocols that are used in todays embedded systems, however, its relation with internal communication and external entities are dependent on web services. An advanced query processing technique suggested by Madden *et al.*, [65] incorporates the design and implementation of acquisitional techniques, however, the proposed method fails to address new query processing methodologies.

A sensing algorithm proposed by Bijarbooneh *et al.*, [66] preserves the energy of active sensor nodes and also provides good data quality so that the load balancing among sensors is achieved. This approach optimizes the overhead of selecting active sensor nodes, however, the factor of data utility increases with the rise in number of sensors. A performance model presented by Xu *et al.*, [67] illustrates how the edge devices connect large network of sensors to the cloud using push/pull optimization technique that is dependent on comparative features of sensor request and sensor data. The proposed approach is effective and energy efficient, nevertheless, the simulation is tested only on few sensors and applications, and it is necessary to conduct tests on massive sensors and applications. Longo *et al.*, [68] presented a method that predicts the performance of public clouds by collecting and codifying the existing techniques and they also formulated several hypotheses that were validated empirically but the effect of multi-tenancy on performance of cloud was not properly analyzed. Yuriama *et al.*,[69] devised a hybrid system that incorporates diverse type of virtual sensors but the actual physical location of sensor is not considered for every request made by the user. An interface is made available to the users, through which they can manage(i.e. addition of new sensors and deletion of old sensors) the deployment environment without the knowledge about intricate details of the implementation and the system is adaptable across different domains of the IoT, however, the work does not provide an in-depth discussion about the actual creation of virtual sensors and their mapping to the actual physical sensors. Perara *et al.,* [70] designed a framework for managing a smart environment(viz., smart wearable, smart home, smart environment, smart city, smart agriculture and smart enterprise) but the proposed solution is yet to be implemented in real time. Madria *et al.,* [71] integrated WSN's and cloud to develop an hybrid data collection system that is capable of monitoring heterogeneous applications and cater to multiple users with varied interests. Here, the network infrastructure is decoupled from the users, enabling widespread dissemination of data that is collected from single WSNs to multiple users, thereby reducing the cost of data acquisition and thus integrating multiple WSNs to develop an IoT application that has multiple data collection requirements. Although, virtualization of sensor cloud eases the management of WSNs, the proposed system generalizes the data collection points and thus is not suitable for construing a domain specific application. Aazam *et al.,* [72] examined the architecture of intercloud computing and discussed issues like storage efficiency, however, a remote cloud is unable to fulfill the user requirements due to rapid increase in digital content.

3.4 Crowdsensing

The technique in which a large group of individuals sense the data and collect information to map, compute and predict any processes of common activity is known as crowdsensing. This technology is also called as community sensing or group sensing, and in IoT it is extensively used to gather the sensor data.

Table 4. Comparison of Different Crowdsensing Techniques

Authors	**Research Concepts**	**Benefits**	**Disadvantages**
Chen *et al.* [73]	Generic framework for crowdsourced live streaming.	Different types of cloud instances and its prices are supported by the proposed design	Number of crowdsourcers and its distribution are highly dynamic in nature.
Jian *et al.* [74]	Novel credible crowdsourcing service provided according to sociality, complexity and mobility of mobile users.	Insights help the mobile user to improve the technical and theoretical point of view.	Service fails to support specific preferences of users within a distributed environment.
Pouryazdan *et al.* [75]	Reputation-unaware approach for crowdsensing utilizes anchor nodes, and it is trustable by stakeholders	No need to store historical data; reputation calculation is done via a distributed approach	Trustworthiness is evaluated efficiently by the non-anchor node compared to that of an anchor node.
Antonic *et al.* [76]	Sensor data in a mobile crowdsensing ecosystem is acquired through a middleware.	Data collection is controlled and limited resources are utilized, thus energy is saved.	Sensor data is filtered before it is being transmitted to the cloud.
Yangui *et al.* [77]	An architecture that favours an hybrid Cloud/fog environment.	Motivating use cases for systematic implementation of IoT applications	Components pertaining to an application are not optimally placed.

Table 4 illustrates a comparison between few crowdsensing techniques.

A framework which was presented by Chen *et al.,* [73] addressed the attributes like location and streaming quality of cloud, however, it failed to provide an optimised solution due to dynamic cloud leasing. Jian *et al.,* [74] illustrated that a crowdsensing service model is dependent on mobile cloud system and it also relies on the mobility, sociality and complexity of mobile users. Crowdsourcing service model is necessary and vital to analyze the characteristics of mobile nodes, service mode decision and service preference mining, however, the proposed method fails to provide diverse, standard and customisable mobile sensing service.

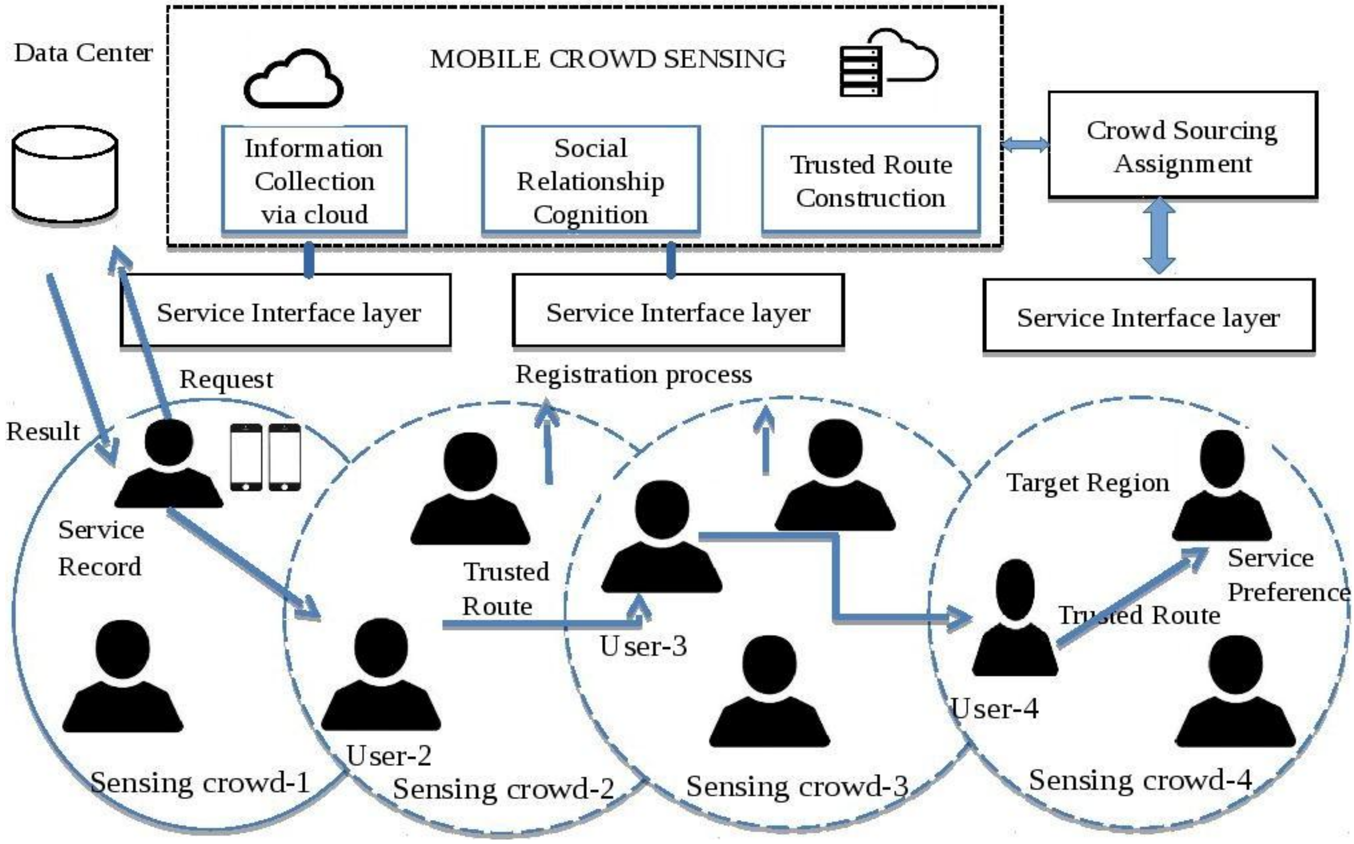

Figure 6: Crowdsensing Model for IoT

Pouryazdan *et al.*, [75] demonstrated how anchor nodes are utilized in crowdsensing through simulations. A set of smartphone users are selected and their corresponding sensing schedules and sensing tasks are assigned without the necessity of storing the historical data, however, an anchor node is incapable of assessing the trustworthiness of remaining nodes when compared to that of non-anchor nodes. Figure 6 depicts an architecture of a crowdsensing model that includes various stages in crowdsensing like information collection, cognition of social relationship, construction of trusted route between users and data centre and crowd sourcing alignment. Mobile service interface regulates the events that occur between a community or crowd and data centre. A Cloud-based middleware that is used in various crowdsensing applications for acquiring sensor data from mobile devices was suggested by Antonic *et al.*, [76] in order to facilitate the sensor data on mobile devices to be filtered before it is transmitted to cloud. This approach is adequate for mobile IoT environment and it helps in flexible management of the sensing process as it controls the network traffic and decreases energy consumption of smart devices. It also facilitates controllable and flexible data collection, however, crowdsensing services are benefited by the functionalities that are offered within an ecosystem.

Sami Yangui *et al.*, [77] introduced an architecture for a Cloud, in which the requirements were derived from several use cases of IoT applications. The proposed method fails to guarantee SLA/QoS and there is a need for sophisticated algorithms as the optimal positioning of application components in a fog and cloud environment is not a straight forward approach. Wang *et al.*, [78] proposed an approach that deals with allocating a portion of work to every shareholder in each cycle. Here, certain aspects of multitask allocation frameworks like complex constraints, privacy preserving mechanisms and heterogeneity of multiple tasks need to be improved. The proposed approach outperforms the baseline methods, however, the user is burdened due to switching between different sensing tasks. An approach which is based on compressive sensing was designed by Kong *et al.*, [79] to reconstruct huge missing data. This approach optimises space utilisation but time utilisation remains unaffected due to the elements lost in the life-time of a node. A scalable method to solve the problem of optimising a cost function for WSNs was suggested by Palopoli *et al.*, [80] but the technique is generic and it is not scalable with the size of network. Moreover the integration of communication architecture with protocol layer is also not supported, and realistic sensor models are not used in this approach.

Tham *et al.*, [81] designed a crowdsensing model in which the people capture information from sensors and share it with others, however, the metrics like quality of contributed services are characterised by the timeliness and information quality of specific real-time sensed quantity. Gulisano *et al.*, [82] presented an efficient stream processing engine which uses a parallelization technique to split queries into subqueries, and it is suitable for processing large data streams. Here, the usage of computational resources is minimised with the combination of elasticity and dynamic load balancing, and the transparent parallelization is provided by stream cloud to preserve the semantics and syntax of centralised queries.Li *et al.,* [83] developed a sensing IoT framework in which the nodes are capable of monitoring, transmitting and storing the sampled data in the framework, however, the performance of system is evaluated using data sets that are acquired by real time deployment.

Sun *et al.,* [84] developed a system for sensing in order to facilitate the users to measure, sense and upload data by using electronic devices. System integration, data quality, information extraction and interaction between system and the participants are some of the challenges faced by those information systems. The problem of information distribution in vehicular clouds was addressed by Talebifard *et al.,* [85] using the idea of semantic based networking of information, and the reliability and efficiency of information distribution was enhanced using selective network coding technique. This approach assists the vehicular cloud in crowd sensing, nevertheless, the forwarding decisions are dependent on the spectral characteristics of cluster and semantics of information. Jiang *et al.,* [86] introduced a remote sensing service description model that is based on ontology in order to provide semantic sensing information but this approach is used only to improve the service discovery and composition.

3.5 Mobile Sensing Techniques and Strategies for IoT

Mobile sensing deals with using the sensors of mobile devices to acquire data from the environment such that those mobile devices provide a promising platform to monitor changes in the lifestyle. UbiFit is an example of a system that's built on mobile sensing platform to automatically detect the duration of an activity and it uses a sensing device and mobile phone in order to facilitate varied and regular physical activity. The components that run on user's smartphone are; an interactive application that is utilized to monitor physical activities and a display unit to abstract and represent the physical activities that the user performs every week.

Table 5. Comparison of Different Mobile Sensing Strategies

Authors	Research Concepts	Benefits	Disadvantages
Perera *et al.* [87]	Processing and collecting the sensor data via a plugin based IoT middleware without any explicit programming.	Expenses incurred on data acquisition, processing requirements and data storage requirements are minimal.	Cannot be programmable whenever it is required
Han *et al.* [88]	Introduced the notion of mobile cloud computing	Integrating social sensing data and mobile sensing data helps to observe and understand the real world.	Data centric sensing applications are dependent on bandwidth of the network.
Flores *et al.*[89]	Systemic client-server approach for analyzing the components of a generic code offloading architecture.	Server is utilized efficiently and an auto scaling mechanism is implemented within the framework.	Does not support multiple levels of granularity and multi-tenancy.
Hong *et al.* [90]	A prototype to integrate sensor networks and ERP's on cloud.	The possibility of loosing any crucial data can be avoided.	Prototype relies on a trust-worthy cloud.
Jayaraman *et al.* [91]	Middleware in a distributed environment to operate on smartphones by collecting and sharing the data.	Task of an application developer is reduced as the middleware is scalable across all generic platforms	Not deployed and evaluated in a real-world application.

Table 5 depicts the comparison between different existing Mobile Sensing strategies. Perara *et al.,* [87] designed a middleware for processing and detecting data without the necessity of programming. This technique facilitates an effective crowd sensing functionality and the cost of sensor data acquisition, data storage requirements and processing requirements can be reduced by this approach. Han *et al.,* [88] demonstrated that the fusion of social sensing data and mobile sensing data gives a more specific and complete observation of the physical world. Here, the sensor data assists the users in understanding the scenario of sensing but the bottleneck for data centric sensing applications (*i.e* network bandwidth) is not handled. Huber Flores *et al.,* [89] adopted an approach to analyse the components of cloud offloading architecture in which the framework resembles a client-server model, according to which the server entity is located in the cloud whereas the client entity is located on the smartphone. The proposed approach uses an autoscaling technique that enables the server to scale horizontally in the cloud but it does not support multiple levels of multi-tenancy and granularity.

International Journal of Applied Engineering Research ISSN 0973-4562 Volume 14, Number 6 (2019) pp. 1220-1243
© Research India Publications. http://www.ripublication.com

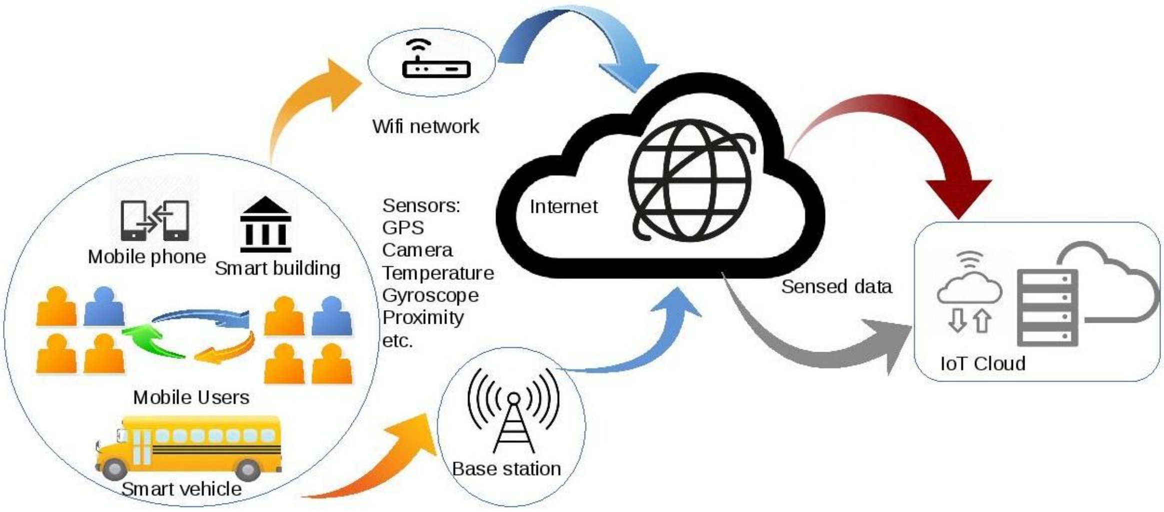

Figure 7: Mobile Sensing Model for IoT

Figure 7 depicts an architecture of a mobile sensing model in which the sensors of mobiles are used as IoT sensors and the sensor data extracted from those sensors are saved in the IoT cloud which can further be processed, analyzed and retrieved whenever it is required by an IoT application.

A design that integrates sensor networks and ERP's in cloud environment through efficient sensor deployment was presented by Hong *et al.,* [90]. The proposed technique minimises the probability of loosing important data or events and the usage of commserver solves the problem of slow remote communication in a cloud environment, however, as the optimization is based on decision making system and ERP, it is not reliable. A collaborative mobile sensing framework presented by Jayaraman *et al.,*[91] is capable of operating on smart devices by acquiring and sharing the sensor data amidst distributed users and applications. The proposed framework has component based design and it promotes reusability of sensing application deployment and it also distinguishes the application specific processing entities from storing, sharing and sensing entities, however, the framework is not deployed and evaluated in a real world application.

A platform developed by More *et al.,* [92] helps to achieve cooperative sensing. The details of communication, processing and network are hidden but all the occurrence of events are visible. This method supports distributed query executions and it can analyse the system performance but the delivery of locations to the cloud server requires more hops, thus the transmission delay gradually increases. A system to control cloud assisted collaborative sensing proposed by Sheng *et al.,* [93] has the potential to reduce energy that is consumed by sensing in smart phone based sensing applications, however, under realistic assumptions, only functional and heuristic algorithms are utilized to discover the sensing schedules. Zhang *et al.,* [94] suggested an approach to protect participants privacy by delinking data from its sources. The proposed method optimises the efficiency of data aggregation, it also allows aggregator to efficiently compute arbitrary aggregation and it guarantees the authenticity of sensing data, however, an untrusted data aggregator is permitted to gather sensed data periodically from mobile users without the prior knowledge of data source.

Wang *et al.,* [95] introduced a method for a neighbour discovery system according to which the inputs from sensor devices are utilized to derive the information. This approach uses principal component analysis method in order to integrate data, however, the acquired data easily becomes outdated and unreliable after a certain time. Agarwal *et al.,* [96] proposed an utility driven middleware to execute community-driven sensing task through which the data and computation can be reused and the middleware is dependent upon the demands of the user and application, however, analytics oriented logic and the task specification language cannot be integrated using this approach. Abdelwahab *et al.,* [97] suggested a multilayer architecture that supports data access in real time, data sharing from global resource, elastic resource provisioning and scaling. The proposed method also shows how the information and cloud resources are shared and utilized widely by the users, however, it leads to the decentralization of data collection and sensing. Su *et al.,* [98] designed a framework to allocate resources among entities like cloud, brokers and mobile users. Here, the cost for a resource can be determined by the cloud and size of cloud is figured by the broker, however, the mobile users are free to select the good strategies in order to attain optimal utility. The heterogeneity in convergent computing and networking was explained by Sanaei *et al.,* [99] and their findings reveal that the sensor nodes are taxonomized and analyzed to fit with the hardware, platform, network and API, however, lack of advanced technologies and absence of standard network protocols decelerate ubiquitous computing and it also hinders the growth of mobile cloud computing.

4. OPEN ISSUES

The various risks in designing a Sensing model [100] are categorised as follows:

1. Technology issues: Technology helps in developing a sensing model and the model is built on an IoT

infrastructure such that it facilitates huge number of sensors and their corresponding data streams. Designing a middleware solution that can handle and manage the demands of several users is a considerable challenge that should be addressed and also there is a need to significantly improve the bandwidth of data communication within an existing infrastructure.

2. Configuration of Sensors: Different aspects of configuration that include intermediate devices, software and sensor embedded software are hidden to protect the data privacy and integrity. The behaviour of a sensor requires to be configured as per the negotiation between sensor owner and sensor consumer whereas the characteristic features of Sensor Publishers and Extended Service Providers should be configured according to cloud middleware. In a distributed system such as SaS, ensuring the interoperability between variety of sensor hardwares and cloud solutions is a challenge, and its architecture does not comply with common standards in key areas. Resource restricted nature of sensors leads to conservation of energy across all entities in the model. To facilitate the design of optimized sensing schedules as well as the techniques that sustain IoT infrastructure, it is crucial to capture the context information. The key to interoperability is standardization and the main motive behind SaS is to ensure that the data is reliable, conceivable and precise. Sensitive information pertaining to sensors such as location must be regulated at both hardware and software levels.

3. Economical issues: The model of SaS should be capable of gathering an enormous data, however, several users who utilize such data lack technical expertise to manage the data.

4. Social issues: To adapt the SaS model, trust and social acceptance are vital such that if there arises a situation where sensor owners do not trust the sensing model, then it might lead to the failure of entire system. According to government and business policies, safety and privacy must be administered at technological level by stringent legitimate terms and conditions. The policies are set to validate and monitor all parties involved in the model in order to facilitate accessibility for the sensor data consumers. The model could be utilized to the maximum by both the sensor owners and data consumers. Since most of the sensor owners are endorsed by wide community, there is a need to automate the process of sensor configuration to enhance the usability.

5. Design issues: The three major problems that are encountered while designing the SaS mechanism are:
   - Non availability of testbed with huge number of sensors.
   - The existing testbeds cover only a specific small portion of area.
   - The available testbeds do not provide any interface such that it can be reused by another SaS solutions.

5. LESSONS LEARNED

In the previous sections, the concepts related to various sensing frameworks were compared, generic techniques for mobile sensing were also reviewed, different methodologies for integrating sensor data and cloud on a middleware were also discussed along with their advantages and disadvantages. This particular section deals with the explanation of characteristic features, shortcomings, challenges and future directions of research with respect to different perspectives of sensing frameworks, existing middleware solutions, cloud and IoT integration mechanisms, various crowdsensing techniques and mobile sensing strategies.

5.1 Sensing Frameworks

In this subsection, we have identified certain attributes of a sensing framework that are based on the analysis of all the literatures pertaining to sensing frameworks. Then we discuss the limitations of existing frameworks, further a list of challenges that are encountered in designing a platform for sensing are explained, and possible solutions are also suggested to overcome the drawbacks of existing methodologies.

• Characteristic features :

a) Data collection: It is the technique of collecting, processing and analyzing data that is originated from billions of IoT devices.

b) Virtualization of network sensors: It deals with the creation of virtual sensors in order to share a single physical instance of a resource among multiple users.

c) Data aggregation: It is the mechanism of expressing the data in an efficient manner by reducing the traffic congestion and power consumption, and increasing data accuracy and network life. It can also be described as the process of integrating the sensor data by adding a value or an attribute to the raw data so that it can be utilized for other purposes.

d) Distributed framework: Adopting to such a framework helps in accurate estimation of the participants contribution.

e) Event detection: It is a process that aims to identify instances of particular type of events.

f) Object recognition: It is the potential of a sensing framework to perceive an object’s physical attributes and recognize various entities.

g) Sensor selection: It is an important step in IoT deployment and the problem of sensor selection is optimally solved by integrating cloud and IoT.

h) Multimedia streaming: It is a mechanism for transmission of data in order to process it as a steady and continuous stream.

i) Interoperability: It means interconnectivity between the devices and networks. As the number of networks and sensor devices increase, the interoperability between them becomes an issue.

j) Sensor discovery and configuration: Detecting a sensor and configuring its hardware facilitates the software platforms to retrieve the data from sensors whenever it is required.

• Shortcomings

a) Data collectors fail to address the issues like user's privacy and computation time.

b) Virtualization permits several applications to execute simultaneously on a single infrastrucure but it leads to low recovery cost.

c) The network based rule engines that are used in aggregation of data is subjected to issues like scalability and it cannot be deployed on a large scale.

d) Desiging a distributed framework is subjected to certain energy and budget constraints.

e) In event detection, the participants are selected by considering factors like quality of interest, mobility patterns and user incentive requirements.

f) The framework for object recognition is dependent on architectural model of IoT for sensing heterogeneous data.

g) Context-aware sensor selection method is based on credentials of sensor attributes and the mechanism of publish services.

h) It consumes more energy from both the mobile clients and cellular operators.

i) Metadata related to sensing platform and sensor data are disassociated and stored separately.

j) Interaction and connectivity between hardware and cloud-based IoT middleware can only be established by configuring the sensors.

• Challenges

(a) The process of data collection require more resources like hardware, power, memory and processors.

(b) Due to virtualization of sensors, the standalone components and databases that are connected to cloud are at high risk.

(c) Aggregation of conceptual data is a great overhead.

(d) Factors like quality of interest, mobility patterns and user incentive requirements must be monitored efficiently by the framework.

(e) There is a need for solution that ensures quality of sensory data and reduce the user disturbance. Continuous sensor data stream is also exposed to problems such as novelty detection.

(f) In an object recognition technique, data is exchanged based on delay and quality requirements.

(g) Selecting the appropriate sensors among huge set of accessible sensors based on various sensor search and selection methods is a overhead.

(h) Handling the multimedia streaming capability is an issue that need to be addressed.

(i) Integration and reuse of data must be facilitated by the sensing framework.

(j) Factors like sampling rate, number of sensors, dynamicity, context and data acquisition must be effectively monitored in order to optimise sensing and data communication.

• Future directions

(a) Data collection points must have access to the cloud and edge so that the data can be analyzed and visualized before processing and it should possess the ability to interact with social media, web services and other devices.

(b) Virtualization must guarantee that the resources are allocated in a flexible and an efficient manner.

(c) Use of an intelligent IoT gateway to preprocess the data at edge can further improve the process of data aggregation.

(d) A distributed framework must allow the end users to define and select the devices in order of hierarchy.

(e) The process of event detection should be improved in a distributed manner and semantic web technologies helps in contextualizing the sensor data.

(f) It is crucial to recognize the objects in real time because integrating an IoT service into real-time object detection helps to expand the service provider's profit and optimize the network resources.

(g) The process of sensor selection helps in intelligent selection of relevant sensors for users depending on the requirement or necessity of users and it also facilitates to acquire long network lifetime.

(h) Multimedia streaming can be effectively monitored by designing the parameters of main system depending upon the tariffs of the business model.

(i) It is necessary to develop an effective data model that supports interoperable and scalable sensor data. The stakeholders should design standards and adapt common protocols in framework so that the overhead of interoperability can be addressed.

(j) Discovery of sensors and their configuration should be automatically monitored by utilising key factors like sensor count, number of on demand schedules, heterogeneity, dynamicity and data acquisition methods.

5.2 Middleware for SaS

This subsection explains few attributes of a middleware. It also conveys the limitations of existing middleware solutions. Those important attributes of middleware for sensing are identified based on the study of literature in the domain of middlewares that are suitable for providing sensing services. Further, the list of challenges that are faced in designing an ideal platform for sensing are discussed, and some solutions are also provided to overcome the shortcomings of present technologies.

• Characteristic features

(a) Data processing: The middleware platform should process large volume of data sensors, applications and devices so that necessary actions can be initiated for real-time applications.

(b) Resource discovery: It deals with the mechanism of discovering the resources such as sensors, metadata and IoT objects.

(c) Data aggregation: A suitable IoT gateway is required to aggregate the data that is collected by sensors.

(d) Data filtering: This task is also performed by the IoT gateway to reduce the cost of transmission, storage and processing.

• Shortcomings

(a) Existing middlewares do not support efficient data aggregation and filtering.

(b) Selection of transition matrix and communication range of sensor devices is influenced by the computation time.

(c) The process of data aggregation is dependent on a cloud platform.

(d) Lack of efficient data upload policies leads to data distortion in IoT networks.

• Challenges

(a) Middleware must provide relevant information about data by decreasing the cost incurred for communication, processing and storage in the cloud.

(b) The process of searching and recruiting suitable service providers for sensing need to be addressed.

(c) Gathering information from various sensors and aggregating it is more crucial to draw insights from the collected data.

(d) Providing intelligent services to users for analyzing data and delivering the data with integrity to the servers are some of the issues that should be addressed.

• Future directions

(a) Virtualize different types of infrastructure-services for storage, computation and sensing.

(b) Sensors must be located by estimating eventual positions of service providers depending on their network interaction.

(c) The raw data that is collected by sensors can be combined on an intermediate edge layer instead of aggregating it on a cloud platform.

(d) Data that contains events of interest must be uploaded to the cloud and edge for processing.

5.3 IoT-Sensor and Cloud Integration

The discussion in this subsection is subjected to the vital factors that are responsible for integration of cloud and IoT. Those factors are recognised based on the literature of cloud and IoT integrations. Then we present the limitations of existing integrated solutions and also discuss challenges that are encountered in designing a platform for integration. Later on, certain directions to resolve the limitations of present technologies are suggested.

• Characteristic features

(a) Data collection: A cloud platform stores and processes the IoT data that is gathered from various devices, sensors and applications.

(b) Sensor-cloud Virtualization: It eases the management of different IoT networks.

(c) Sensing Architecture: A modularized architecture makes the system robust, reliable, flexible and automated.

(d) Data quality: The quality of collected data must not be degraded, rather it should be checked during the occurrence of progressive events.

(e) Interconnectivity: Natively parallel architectures like publish-subscribe model supports the IoT devices to be interoperable when an event occurs within a stipulated time.

(f) Data integration: A cloud based solution or protocol considers spatiotemporal features of data in order to integrate different data.

• Shortcomings

(a) Decoupling the network infrastructure from the users, enables widespread dissemination of data.

(b) Virtualizing sensors is not suitable for construing a demand based domain specific application.

(c) Accessing manufacturing resources and realisation of intelligent perception and capabilities are the bottlenecks of a sensing platform.

(d) With the increase in count of sensors, the sum of correlation between a node and selected sensor becomes a large factor in data utility.

(e) Devices fail to interoperate with solutions provided by different vendors.

(f) Integation of data reduces the energy consumption required for collecting data.

• Challenges

(a) Handling the safety of data and managing the energy constraints of micro controller are some of the problems that need to be addressed or resolved.

(b) Managing different types of IoT networks and integrating them are the problems that need to be addressed.

(c) Cloud sensing platform is subjected to sensor energy and budget constraints.

(d) Selection of a feasible set of active sensor nodes is necessary for maintaining the quality of data.

(e) Assuring security and scalability across smart city applications.

(f) Link quality and spatiotemporal correlation of data are the only factors that are considered in designing the protocol.

• Future directions

(a) Development of an hybrid data collection system that is capable of monitoring heterogeneous applications.

(b) Hybrid push-pull based strategies must be adopted to enable the communication between edge and sensor layers.

(c) System must be adaptable across different domains of the IoT.

(d) The problem of choosing feasible set of sensor nodes must be optimized using load balancing among sensors and by implementing machine learning based techniques.

(e) Plugin based IoT middleware that is built on generic platform should gather and process the sensor data such that it can work on resource constraint IoT devices.

(f) The performance of designed protocol must be evaluated in terms of data prediction accuracy, data utility and energy efficiency.

5.4 Crowd sensing Technique

In this subsection, we discuss certain characteristics related to crowdsensing which are identified based on the literature of crowdsensing paradigms. Then we present the limitations of those existing technologies. Later, few challenging factors that need to be considered in designing effective solutions for crowdsensing are conveyed and certain possible solutions to overcome the drawbacks of existing system are also suggested.

• Characteristic features

(a) Streaming quality: Due to the generation of huge data pertaining to IoT, there is a possibility of decline in the quality of streamed data but a cost effective optimised cloud service could improve the quality of streamed data.

(b) Service migration: The different types and varying prices of cloud instances are accommodated in real time.

(c) Information distribution: Diversifying data in IoT architecture has an impact on stakeholders, infrastructure and analytics. Utilizing a distributed cloud for transfer of IoT data helps in processing the data efficiently.

(d) Sensing capability: The ability to acquire knowledge from IoT objects helps to perform various tasks.

• Shortcomings

(a) Lack of cost effective optimizing techniques to enhance the streaming quality of IoT data.

(b) Distribution process of the crowdsourcers is highly dynamic in nature.

(c) Centralized processing and collection of data may overlap and create undesirable effects in certain IoT scenarios.

(d) Requires active involvement or participation of individuals in order to contribute for the sensor data.

• Challenges

(a) The user is burdened due to switching between different sensing tasks. Information extraction and interaction between system and the participants are some of the challenges that need to be addressed.

(b) Mobility, sociality and complexity of mobile users are the overheads.

(c) Managing and organising highly distributed data is an important challenge. Integration of the system and maintaining the data quality are the factors that need to be addressed.

(d) Limitation of resources and data integrity are the challenging factors.

• Future directions

(a) Privacy preserving mechanisms and heterogeneity of multiple tasks in data streams should be improved.

(b) Storing the historical data is not essential but the timeliness and information quality of specific real-time sensed data must be maintained.

(c) Selective network coding technique and other enhanced methodologies must be used to improve the idea of semantic based networking, and the reliability and efficiency of information distribution should also be enhanced. Remote sensing service description model that is based on ontology of semantic sensing information can also improve the intelligent service discovery.

(d) Opportunistic sensing must be automated and involvement of the users should be minimized.

5.5 Mobile sensing technologies and Strategies

In this subsection we identify certain attributes related to mobile sensing that are identified by the literature of smartphone sensing. Then the limitations of mobile sensing technologies are notified and challenging factors governing the mobile sensing platform are systematically discussed. Further, effective solutions to overcome the drawbacks of existing techniques are also explained.

• Characteristic features

(a) Data acquisition: It is an automated process or activity of gathering and organizing the source data from sensors so that the collected information can be sampled, stored, retrieved and manipulated depending upon the type of application.

(b) Heterogeneity: The term heterogeneity, in general denotes diversity or dissimiliarity. As IoT aims to connect many heterogeneous devices to provide advanced applications, there’s a need of platform which can facilitate and control the data that’s collected from those devices.

(c) Multitenancy: It is a variant of computing architecture that depicts the method in which the data is organised in the application and it also permits multiple users to work in a software environment simultaneously using their own user interface, services and resources.

(d) Granularity: It is the ability to modify and display the things into a finer level of detail such that it can be managed effortlessly.

(e) Resource allocation: It is the mechanism of accommodating the workload of all applications by allocating the necessary resources to ensure that the expected results are met.

(f) Data aggregation: It is an efficient mechanism of collecting the data in order to improve network lifetime, energy consumption and data accuracy.

• Shortcomings

(a) Automated system for data collection requires more energy.

(b) Lack of advanced technologies and non availability of standard network protocols for heterogenous computing.

(c) Task of predicting the performance is not uniformly articulated in the cloud.

(d) Granularity is driven by capability or business need, it depends on the design and size of application and it is influenced by the deployment domain.

(e) Cloud services and resources does not contain different levels of abstraction for end developers or users.

(f) The efficiency of data aggregation relies on size of the sensing data and design of the network. The untrusted data aggregators can collect sensed data at regular intervals from mobile users without the knowledge of data source.

• Challenges

(a) There is a need to minimize the cost of data acquisition and sign sparsity induced by network spectrum

(b) Handling the heterogeneous data that’s collected by IoT sensors is a major hurdle.

(c) Multitenancy effects the performance of different sensing providers due to substantial differences in computing and it is also essential to increase the efficiency with multi-tenant cloud software architectures.

(d) Characterizing the level of detail in a set of data must be improved.

(e) The factors like scalability, context awareness and heterogeneity of hardware and software plays a crucial role and they also challenge the resource management.

(f) It is very necessary to optimise the efficiency of data aggregation.

• Future directions

(a) The process of data collection and sensing must be decentralized to avoid the sparsity induced by the network bandwidth.

(b) Heterogeneity of sensor nodes must be organised and analysed to ensure that it fits with the hardware, platform and API.

(c) The services to users must be delivered quickly without the requirement of additional resources thereby allowing the application to be managed from a central location.

(d) Granularity must be induced in computing tasks to work in the context of a complete scenario.

(e) The interactions among users and devices must be leveraged by particular contexts.

(f) The authenticity of sensing data must be guaranteed to the end users.

6. CONCLUSIONS

The idea of sensing as a service is regarded as a vital factor in the design of a middleware for IoT and it has a wide scope for inter-operability, since it has the potential to exchange non-proprietary data within an IoT environment. It has more number of challenges like selective transmission of data and maintaining security of sensor data. There is a need for a middleware to control and operate the vast number of IoT devices and the user need not handle the physical sensing infrastructure instead he/she can take the advantage from existing one. Cloud has the potential to store data from sensing devices and leverage its performance. All the IoT devices can rely on a cloud for storage, computation and handling effective sensing operations. It’s possible to visualize and analyse large sensor data efficiently using the ideology of sensor cloud integration.

6.1 Acknowledgement

The authors would like to thank all the faculty and research scholars of the Department of Computer Science and Engineering, University Visvesvaraya College of Engineering, Bangalore University for providing all the resources and necessary guidance to carry out this research work.

REFERENCES

[1] S. Pattar, R. Buyya, K. R. Venugopal, S. S. Iyengar, and L. M. Patnaik, “Searching for the IoT Resources: Fundamentals, Requirements, Comprehensive Review and Future Directions,” *IEEE Communications Surveys and Tutorials*, vol. 20, no. 3, pp. 2101–2132, 2018.

[2] A. Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari, and M. Ayyash, “Internet of Things: A survey on Enabling Technologies, Protocols, and Applications,” *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2347–2376, 2015.

[3] J. Fenn and M. Raskino, “Gartners Hype Cycle Special Report for 2011, Gartner Inc.” https://www.gartner.com/doc/1754719/hype-cycle-emerging-technologies , 2011.

[4] R. Mafrur, I. Nugraha, and D. Choi, “Concept, Design and Implementation of Sensing as a Service Framework,” *ACM 9th International Conference on Ubiquitous Information Management and Communication*, pp. 86–89, 2015.

[5] B. Arunachalam, D. Arjun, P. R. BB, H. Pasupuleti, and V. Dwarakanath, “Sensing Service Framework for Climate Alert System using WSN-Cloud Infrastructure,” *IEEE 9th International Conference on Sensing Technology (ICST)*, pp. 671–676, 2015.

[6] S. Manjula, C. Abhilash, K. Shaila, K. Venugopal, and L. Patnaik, “Performance of Aodv Routing Protocol Using Group and Entity Mobility Models in Wireless Sensor Networks,” *Proceedings of the International MultiConference of Engineers and Computer Scientists*, vol. 2, pp. 1212–1217, 2008.

[7] S. Tarannum, S. Srividya, D. Asha, R. Padmini, L. Nalini, K. Venugopal, and L. Patnaik, “Dynamic Hierarchical Communication Paradigm for Wireless Sensor Networks: A Centralized, Energy Efficient Approach,” *11th IEEE Singapore International Conference on Communication Systems, (ICCS).*, pp. 959–963, 2008.

[8] M. M. Hassan, B. Song, and E. N. Huh, “A Framework of Sensor-Cloud Integration Opportunities and Challenges,” *Proceedings of the ACM 3rd international conference on Ubiquitous information management and communication*, pp. 618–626, 2009.

[9] X. Sheng, J. Tang, X. Xiao, and G. Xue, “Sensing as a Service: Challenges, Solutions and Future Directions,” *IEEE Sensors journal*, vol. 13, no. 10, pp. 3733–3741, 2013.

[10] S. Chattopadhyay and A. Banerjee, “Algorithmic Strategies for Sensing-as-a-Service in the Internet-of-Things Era,” *IEEE/ACM 8th International Conference on Utility and Cloud Computing (UCC)*, pp. 387–390, 2015.

[11] D. Niyato, D. I. Kim, P. Wang, and L. Song, “A Novel Caching Mechanism for Internet of Things (IoT) Sensing Service with Energy Harvesting,” *IEEE International Conference on Communications (ICC)*, pp. 1–6, 2016.

[12] M. Fazio, M. Villari, and A. Puliafito, “Sensing Technologies for Homeland Security in Cloud Environments,” *IEEE Fifth International Conference on Sensing Technology (ICST)*, pp. 165–170, 2011.

[13] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions,” *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[14] B.C. Chifor, I. Bica, and V.V. Patriciu, “Sensing service architecture for smart cities using social

network platforms," *Soft Computing*, vol. 21, no. 16, pp. 4513–4522, 2017.

[15] J. Sang, T. Mei, and C. Xu, "Activity Sensor: Check-in Usage Mining for Local Recommendation," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, no. 3, p. 41, 2015.

[16] B. Zhang, Z. Song, C. H. Liu, J. Ma, and W. Wang, "An Event-Driven QOI-Aware Participatory Sensing Framework with Energy and Budget Constraints," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 6, no. 3, p. 42, 2015.

[17] J. Wan, C. Zou, K. Zhou, R. Lu, and D. Li, "IoT Sensing Framework with Inter-Cloud Computing Capability in Vehicular Networking," *Electronic Commerce Research*, vol. 14, no. 3, pp. 389–416, 2014.

[18] N. Ghosh, D. Chatterjee, and S. K. Ghosh, "An Efficient Heuristic-Based Role Mapping Framework for Secure and Fair Collaboration in SaaS Cloud," *IEEE International Conference on Cloud and Autonomic Computing (ICCAC)*, pp. 227–236, 2014.

[19] A. E. Al-Fagih, F. M. Al-Turjman, W. M. Alsalih, and H. S. Hassanein, "A Priced Public Sensing Framework for Heterogeneous IoT Architectures," *IEEE Transactions on Emerging Topics in Computing*, vol. 1, no. 1, pp. 133–147, 2013.

[20] P. H. Tsai, Y. J. Lin, Y. Z. Ou, E. T. H. Chu, and J. W. Liu, "A Framework for Fusion of Human Sensor and Physical Sensor Data," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 44, no. 9, pp. 1248–1261, 2014.

[21] D. Philipp, F. Durr, and K. Rothermel, "A Sensor Network Abstraction for Flexible Public Sensing Systems," *IEEE 8th International Conference on Mobile Adhoc and Sensor Systems (MASS)*, pp. 460–469, 2011.

[22] Y. C. Hsu, C. H. Lin, and W. T. Chen, "Design of a Sensing Service Architecture for Internet of Things with Semantic Sensor Selection," *IEEE 11th International Conference on Ubiquitous Intelligence and Computing, 2014 and IEEE 11th International Conference on and Autonomic and Trusted Computing, and IEEE 14th International Conference on Scalable Computing and Communications and Its Associated Workshops (UTC-ATC-ScalCom)*, pp. 290–298, 2014.

[23] K. M. Sim, "Agent-Based Cloud Computing," *IEEE Transactions on Services Computing*, vol. 5, no. 4, pp. 564–577, 2012.

[24] S. Son and K. M. Sim, "A Price-and-Time-Slot-Negotiation Mechanism for Cloud Service Reservations," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 42, no. 3, pp. 713–728, 2012.

[25] Y. Wei and M. B. Blake, "Service-Oriented Computing and Cloud Computing: Challenges and Opportunities," *IEEE Internet Computing*, vol. 14, no. 6, pp. 72–75, 2010.

[26] D. H. Le, H. L. Truong, G. Copil, S. Nastic, and S. Dustdar, "Salsa: A Framework for Dynamic Configuration of Cloud Services," *IEEE 6th International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 146–153, 2014.

[27] Y. Zhang, M. Qiu, C. W. Tsai, M. M. Hassan, and A. Alamri, "Health-CPS: Healthcare Cyber-Physical System Assisted by Cloud and Big Data," *IEEE Systems Journal*, vol. 1, no. 99, pp. 1–8, 2015.

[28] J. Zhang, B. Iannucci, M. Hennessy, K. Gopal, S. Xiao, S. Kumar, D. Pfeffer, B. Aljedia, Y. Ren, M. Griss *et al.*, "Sensor Data as a Service–a Federated Platform for Mobile Data-Centric Service Development and Sharing," *IEEE International Conference on Services Computing (SCC)*, pp. 446–453, 2013.

[29] C. Perera, A. Zaslavsky, C. H. Liu, M. Compton, P. Christen, and D. Georgakopoulos, "Sensor Search Techniques for Sensing as a Service Architecture for the Internet of Things," *IEEE Sensors Journal*, vol. 14, no. 2, pp. 406–420, 2014.

[30] M. Anisetti, C. A. Ardagna, and E. Damiani, "A Certification-Based Trust Model for Autonomic Cloud Computing Systems," *IEEE International Conference on Cloud and Autonomic Computing (ICCAC)*, pp. 212–219, 2014.

[31] S. Abdelwahab, B. Hamdaoui, and Guizani, "Replisom: Disciplined Tiny Memory Replication for Massive IoT Devices in LTE Edge Cloud," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 327–338, 2016.

[32] S. H. Kim and D. Kim, "Enabling Multi-Tenancy via Middleware-Level Virtualization with Organization Management in the Cloud of Things," *IEEE Transactions on Services Computing*, vol. 8, no. 6, pp. 971–984, 2015.

[33] K. Hu, V. Sivaraman, B. G. Luxan, and A. Rahman, "Design and Evaluation of a Metropolitan Air Pollution Sensing System," *IEEE Sensors Journal*, vol. 16, no. 5, pp. 1448–1459, 2016.

[34] E. Patti, A. Acquaviva, M. Jahn, F. Pramudianto, R. Tomasi, D. Rabourdin, J. Virgone, and E. Macii, "Event-Driven User-Centric Middleware for Energy-Efficient Buildings and Public Spaces," *IEEE Systems Journal*, vol. 10, no. 3, pp. 1137–1146, 2016.

[35] L. Braubach, K. Jander, and A. Pokahr, "A Middleware for Managing Non-Functional Requirements in Cloud PaaS," *IEEE International Conference on Cloud and Autonomic Computing (ICCAC)*, pp. 83–92, 2014.

[36] B. Kantarci and H. T. Mouftah, "Sensing Services in Cloud-Centric Internet of Things: A Survey, Taxonomy and Challenges," *IEEE International Conference on Communication Workshop (ICCW)*, pp. 1865–1870, 2015.

[37] M. Fazio, M. Paone, A. Puliafito, and M. Villari, "Huge Amount of Heterogeneous Sensed Data Needs the Cloud," *IEEE 9th International Multi-Conference on Systems, Signals and Devices (SSD)*, pp. 1–6, 2012.

[38] J. Zhou, T. Leppänen, E. Harjula, M. Ylianttila, T. Ojala, C. Yu, and H. Jin, "Cloudthings: A Common Architecture for Integrating the Internet of Things with Cloud Computing," *IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD)*, pp. 651–657, 2013.

[39] R. Lea and M. Blackstock, "City Hub: A Cloud-Based IoT Platform for Smart Cities," *IEEE 6th International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 799–804, 2014.

[40] P. P. Pereira, J. Eliasson, R. Kyusakov, J. Delsing, A. Raayatinezhad, and M. Johansson, "Enabling Cloud Connectivity for Mobile Internet of Things Applications," *IEEE 7th International Symposium on Service Oriented System Engineering (SOSE)*, pp. 518–526, 2013.

[41] B. P. Rao, P. Saluia, N. Sharma, A. Mittal, and S. V. Sharma, "Cloud Computing for Internet of Things & Sensing based Applications," *Sixth International Conference on Sensing Technology (ICST)*, pp. 374–380, 2012.

[42] A. Taivalsaari and T. Mikkonen, "Cloud Technologies for the Internet of Things: Defining a Research Agenda Beyond the Expected Topics," *41st Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pp. 484–488, 2015.

[43] X. Zhu, C. Chen, L. T. Yang, and Y. Xiang, "Angel: Agent-Based Scheduling for Real-Time Tasks in Virtualized Clouds," *IEEE Transactions on Computers*, vol. 64, no. 12, pp. 3389–3403, 2015.

[44] L. Zhang, Q. Wang, and X. Shu, "A Mobile-Agent-Based Middleware for Wireless Sensor Networks Data Fusion," *I2MTC'09. IEEE Instrumentation and Measurement Technology Conference.*, pp. 378–383, 2009.

[45] F. Fargo, C. Tunc, Y. Al-Nashif, A. Akoglu, and S. Hariri, "Autonomic Workload and Resources Management of Cloud Computing Services," *IEEE International Conference on Cloud and Autonomic Computing (ICCAC)*, pp. 101–110, 2014.

[46] M. Aazam and E.-N. Huh, "Resource Management in Media Cloud of Things," *IEEE 43rd International Conference on Parallel Processing Workshops (ICCPW)*, pp. 361–367, 2014.

[47] T. Laukkarinen, J. Suhonen, and M. Hännikäinen, "An embedded cloud design for internet-of-things," *International Journal of Distributed Sensor Networks*, vol. 9, no. 11, p. 790130, 2013.

[48] C. Perera, A. Zaslavsky, P. Christen, A. Salehi, and D. Georgakopoulos, "Capturing Sensor Data from Mobile Phones using Global Sensor Network Middleware," *IEEE 23rd International Symposium on Personal Indoor and Mobile Radio Communications (PIMRC)*, pp. 24–29, 2012.

[49] C. Perera, P. Jayaraman, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Dynamic Configuration of Sensors using Mobile Sensor Hub in Internet of Things Paradigm," *IEEE Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pp. 473–478, 2013.

[50] C. Doukas and I. Maglogiannis, "Managing Wearable Sensor Data Through Cloud Computing," *IEEE Third International Conference on Cloud Computing Technology and Science (CloudCom)*, pp. 440–445, 2011.

[51] M. Yuriyama, T. Kushida, and M. Itakura, "A New Model of Accelerating Service Innovation with Sensor-Cloud Infrastructure," *IEEE Annual SRII Global Conference*, pp. 308–314, 2011.

[52] M. Barcelo, A. Correa, J. Llorca, A. M. Tulino, J. L. Vicario, and A. Morell, "IoT-Cloud Service Optimization in Next Generation Smart Environments," *IEEE Journal on Selected Areas in Communications*, vol. 34, no. 12, pp. 4077–4090, 2016.

[53] S. Nastic, S. Sehic, M. Vogler, H.-L. Truong, and S. Dustdar, "Patricia–a Novel Programming Model for IoT Applications on Cloud Platforms," *IEEE 6th International Conference on Service-Oriented Computing and Applications (SOCA)*, pp. 53–60, 2013.

[54] D. H. Phan, J. Suzuki, S. Omura, and K. Oba, "Toward Sensor-Cloud Integration as a Service: Optimizing Three-Tier Communication in Cloud-Integrated Sensor Networks," *Proceedings of the 8th International Conference on Body Area Networks*, pp. 355–362, 2013.

[55] X. Zheng, P. Martin, K. Brohman, and L. Da Xu, "Cloud Service Negotiation in Internet of Things Environment: A Mixed Approach," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1506–1515, 2014.

[56] C. Wang, Z. Bi, and L. Da Xu, "IoT and Cloud Computing in Automation of Assembly Modeling Systems," *IEEE Transactions on Industrial Informatics*, vol. 10, no. 2, pp. 1426–1434, 2014.

[57] H. L. Truong and S. Dustdar, "Principles for Engineering IoT Cloud Systems," *IEEE Cloud Computing*, vol. 2, no. 2, pp. 68–76, 2015.

[58] L. Roffia, F. Morandi, J. Kiljander, A. D Elia, F. Vergari, F. Viola, L. Bononi, and T. S. Cinotti, "A Semantic Publish-Subscribe Architecture for the Internet of Things," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1274–1296, 2016.

[59] F. H. Bijarbooneh, W. Du, E. C. H. Ngai, X. Fu, and J. Liu, "Cloud-Assisted Data Fusion and Sensor Selection for Internet of Things," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 257–268, 2016.

[60] Y. Xu and A. Helal, "Scalable Cloud–Sensor Architecture for the Internet of Things," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 285–298, 2016.

[61] C. Perera, P. P. Jayaraman, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "MOSDEN: An Internet of Things Middleware for Resource Constrained Mobile Devices," *IEEE 47th Hawaii International Conference on System Sciences*, pp. 1053–1062, 2014.

[62] T. Le Vinh, S. Bouzefrane, J. M. Farinone, A. Attar, and B. P. Kennedy, "Middleware to Integrate Mobile Devices, Sensors and Cloud Computing," *Procedia Computer Science*, vol. 52, pp. 234–243, 2015.

[63] Y. Xu and S. Helal, "Application Caching for Cloud-Sensor Systems," *Proceedings of the 17th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pp. 303–306, 2014.

[64] P. Spiess, S. Karnouskos, D. Guinard, D. Savio, O. Baecker, L. M. S. De Souza, and V. Trifa, "SOA-Based Integration of the Internet of Things in Enterprise Services," *IEEE International Conference on Web Services*, pp. 968–975, 2009.

[65] S. R. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong, "TinyDB: An Acquisitional Query Processing System for Sensor Networks," *ACM Transactions on database systems (TODS)*, vol. 30, no. 1, pp. 122–173, 2005.

[66] F. H. Bijarbooneh, W. Du, E. Ngai, and X. Fu, "Energy-Efficient Sensor Selection for Data Quality and Load Balancing in Wireless Sensor Networks," *IEEE 22nd International Symposium on Quality of Service (IWQoS)*, pp. 338–343, 2014.

[67] Y. Xu, S. Helal, and M. Scmalz, "Optimizing Push/Pull Envelopes for Energy-Efficient Cloud-Sensor Systems," *Proceedings of the 14th ACM international conference on Modeling, analysis and simulation of wireless and mobile systems*, pp. 17–26, 2011.

[68] F. Longo, D. Bruneo, S. Distefano, G. Merlino, and A. Puliafito, "Stack4things: a sensing-and-actuation-as-a-service framework for iot and cloud integration," *Annals of Telecommunications*, vol. 72, no. 1-2, pp. 53–70, 2017.

[69] M. Yuriyama and T. Kushida, "Sensor-Cloud Infrastructure-Physical Sensor Management with Virtualized Sensors on Cloud Computing," *IEEE 13th International Conference on Network-Based Information Systems (NBiS)*, pp. 1–8, 2010.

[70] C. Perera, C. H. Liu, S. Jayawardena, and M. Chen, "A Survey on Internet of Things from Industrial Market Perspective," *IEEE Access*, vol. 2, pp. 1660–1679, 2014.

[71] S. Madria, V. Kumar, and R. Dalvi, "Sensor Cloud: A Cloud of Virtual Sensors," *IEEE software*, vol. 31, no. 2, pp. 70–77, 2014.

[72] M. Aazam and E.N. Huh, "Inter-Cloud Media Storage and Media Cloud Architecture for Inter-Cloud Communication," *IEEE 7th International Conference on Cloud Computing (CLOUD)*, pp. 982–985, 2014.

[73] F. Chen, C. Zhang, F. Wang, J. Liu, X. Wang, and Y. Liu, "Cloud-Assisted Live Streaming for Crowdsourced Multimedia Content," *IEEE Transactions on Multimedia*, vol. 17, no. 9, pp. 1471–1483, 2015.

[74] A. Jian, G. Xiaolin, Y. Jianwei, S. Yu, and H. Xin, "Mobile Crowd Sensing for Internet of Things: A Credible Crowdsourcing Model in Mobile-Sense Service," *IEEE International Conference on Multimedia Big Data (BigMM)*, pp. 92–99, 2015.

[75] M. Pouryazdan, B. Kantarci, T. Soyata, and H. Song, "Anchor-Assisted and Vote-Based Trustworthiness Assurance in Smart City Crowdsensing," *IEEE Access*, vol. 4, pp. 529–541, 2016.

[76] A. Antonic, K. Roankovic, M. Marjanovic, K. Pripuic *et al.*, "A Mobile Crowdsensing Ecosystem Enabled by a Cloud-Based Publish/Subscribe Middleware," *International Conference on Future Internet of Things and Cloud (FiCloud)*, pp. 107–114, 2014.

[77] S. Yangui, P. Ravindran, O. Bibani, R. H. Glitho, N. B. Hadj-Alouane, M. J. Morrow, and P. A. Polakos, "A Platform As-a-Service for Hybrid Cloud/Fog Environments," *IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*, pp. 1–7, 2016.

[78] J. Wang, Y. Wang, D. Zhang, L. Wang, H. Xiong, S. Helal, Y. He, and F. Wang, "Fine-Grained Multi-Task Allocation for Participatory Sensing with a shared budget," *IEEE Internet of Things Journal*, vol. 3, no. 6, pp. 1395–1405, 2016.

[79] L. Kong, M. Xia, X.Y. Liu, G. Chen, Y. Gu, M. Y. Wu, and X. Liu, "Data Loss and Reconstruction in Wireless Sensor Networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 25, no. 11, pp. 2818–2828, 2014.

[80] L. Palopoli, R. Passerone, and T. Rizano, "Scalable Offline Optimization of Industrial Wireless Sensor Networks," *IEEE Transactions on Industrial Informatics*, vol. 7, no. 2, pp. 328–339, 2011.

[81] C.K. Tham and T. Luo, "Quality of Contributed Service and Market Equilibrium for Participatory Sensing,"

IEEE Transactions on Mobile Computing, vol. 14, no. 4, pp. 829–842, 2015.

[82] V. Gulisano, R. Jimenez-Peris, M. Patino-Martinez, C. Soriente, and P. Valduriez, "Streamcloud: An Elastic and Scalable Data Streaming System," *IEEE Transactions on Parallel and Distributed Systems*, vol. 23, no. 12, pp. 2351–2365, 2012.

[83] S. Li, L. Da Xu, and X. Wang, "Compressed Sensing Signal and Data Acquisition in Wireless Sensor Networks and Internet of Things," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 4, pp. 2177–2186, 2013.

[84] Y. Sun and K. Nakata, "An Agent-Based Architecture for Participatory Sensing Platform," *IEEE 4th International Universal Communication Symposium (IUCS)*, pp. 392–400, 2010.

[85] P. Talebifard and V. C. Leung, "Towards a Content-Centric Approach to Crowd-Sensing in Vehicular Clouds," *Journal of Systems Architecture*, vol. 59, no. 10, pp. 976–984, 2013.

[86] L. Jiang and Y. Zhou, "Ontology based Remote Sensing Information Service Intelligent Discovery and Composition," *2nd International Conference on Remote Sensing, Environment and Transportation Engineering (RSETE)*, pp. 1–4, 2012.

[87] C. Perera, D. S. Talagala, C. H. Liu, and J. C. Estrella, "Energy-Efficient Location and Activity-Aware On-Demand Mobile Distributed Sensing Platform for Sensing as a Service in IoT Clouds," *IEEE Transactions on Computational Social Systems*, vol. 2, no. 4, pp. 171–181, 2015.

[88] Q. Han, S. Liang, and H. Zhang, "Mobile Cloud Sensing, Big Data, and 5G Networks Make an Intelligent and Smart World," *IEEE Network*, vol. 29, no. 2, pp. 40–45, 2015.

[89] H. Flores, P. Hui, S. Tarkoma, Y. Li, S. Srirama, and R. Buyya, "Mobile Code Offloading: From Concept to Practice and Beyond," *IEEE Communication*, vol. 53, no. 3, pp. 80–88, 2015.

[90] H. J. Hong, C. L. Fan, Y. C. Lin, and C. H. Hsu, "Optimizing Cloud-Based Video Crowdsensing," *IEEE Internet of Things Journal*, vol. 3, no. 3, pp. 299–313, 2016.

[91] P. P. Jayaraman, C. Perera, D. Georgakopoulos, and A. Zaslavsky, "Efficient Opportunistic Sensing using Mobile Collaborative Platform Mosden," *IEEE 9th International Conference on Collaborative Computing: Networking, Applications and Worksharing (Collaboratecom)*, pp. 77–86, 2013.

[92] S. Mori, Y. C. Wang, T. Umedu, A. Hiromori, H. Yamaguchi, and T. Higashino, "Design and Architecture of Cloud-Based Mobile Phone Sensing Middleware," *IEEE Second Symposium on Network Cloud Computing and Applications (NCCA)*, pp. 102–109, 2012.

[93] X. Sheng, J. Tang, and W. Zhang, "Energy-Efficient Collaborative Sensing with Mobile Phones," *INFOCOM, 2012 Proceedings IEEE*, pp. 1916–1924, 2012.

[94] Y. Zhang, Q. Chen, and S. Zhong, "Privacy-Preserving Data Aggregation in Mobile Phone Sensing," *IEEE Transactions on Information Forensics and Security*, vol. 11, no. 5, pp. 980–992, 2016.

[95] L. Wang, J. Yang, and W. Liu, "Leveraging Participatory Extraction to Mobility Sensing for Individual Discovery in Crowded Environments," *International Journal of Distributed Sensor Networks*, vol. 9, no. 10, pp. 246–916, 2013.

[96] V. Agarwal, N. Banerjee, D. Chakraborty, and S. Mittal, "USense–A Smartphone Middleware for Community Sensing," *IEEE 14th International Conference on Mobile Data Management (MDM)*, pp. 56–65, 2013.

[97] S. Abdelwahab, B. Hamdaoui, M. Guizani, and A. Rayes, "Enabling Smart Cloud Services through Remote Sensing: An Internet of Everything Enabler," *IEEE Internet Things Journal*, vol. 1, no. 3, pp. 276–288, 2014.

[98] Z. Su, Q. Xu, M. Fei, and M. Dong, "Game Theoretic Resource Allocation in Media Cloud with Mobile Social Users," *IEEE Transactions on Multimedia*, vol. 18, no. 8, pp. 1650–1660, 2016.

[99] Z. Sanaei, S. Abolfazli, A. Gani, and R. Buyya, "Heterogeneity in Mobile Cloud Computing: Taxonomy and Open Challenges," *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 369–392, 2014.

[100] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Sensing as a service model for smart cities supported by internet of things," *Transactions on Emerging Telecommunications Technologies*, vol. 25, no. 1, pp. 81–93, 2014.

Authors Profile [1, 2]

Sampath Kumar Y R received his B.E. and M.Tech degree in Computer Science and Engineering from Visvesvaraya Technological University, Belagavi. He is pursuing his Ph.D. at Bangalore University and his current research focuses on cloud computing and IoT.

Champa H N is a Professor and Chairperson at Computer Science and Engineering department, Bangalore University, Bengaluru. She holds a doctoral degree in Computer Science from the University of Mysore. She has about eight international publications to her credit and her research interests include image processing, data mining and Machine learning. She has twenty eight years of teaching experience and is currently working in the area of handwriting recognition.