A Survey on Mainstream Dimensions of Edge Computing

YUANDA WANG, Department of Computer and Information Science and Engineering, University of Florida, USA
HAIBO WANG, Department of Computer and Information Science and Engineering, University of Florida, USA
SHIGANG CHEN, Department of Computer and Information Science and Engineering, University of Florida, USA
YE XIA, Department of Computer and Information Science and Engineering, University of Florida, USA

Driven by the booming of Internet of Things and 4G/5G communications, an increasingly large number of edge devices, e.g., sensors and cell phones, are continuously producing data service requests, which should be processed in high quality. Recent years have seen a paradigm shift from centralized cloud computing toward edge computing. Edge computing is a distributed computing paradigm that utilizes computing and storage resources of edge devices. Compared with traditional cloud computing, edge computing migrates data computation and storage to the edge devices. Recently many technical breakthroughs have been made in edge computing. This survey reviews existing research on edge computing with a focus on the three mainstream dimensions: resource allocation, data fusion and security. We present specific techniques of the three dimensions and how they can contribute to the improvement of edge computing. Emerging and prospective application fields that would benefit from edge computing are also discussed.

CCS Concepts: • **Networks** → **Network design principles**.

Additional Key Words and Phrases: edge computing, cloud computing, resource allocation, data fusion, security

ACM Reference Format:
Yuanda Wang, Haibo Wang, Shigang Chen, and Ye Xia. 2021. A Survey on Mainstream Dimensions of Edge Computing. In *2021 the 5th International Conference on Information System and Data Mining (ICISDM 2021), May 27–29, 2021, Silicon Valley, CA, USA*. ACM, New York, NY, USA, 13 pages. https://doi.org/10.1145/3471287.3471295

ICISDM 2021, May 27–29, 2021, Silicon Valley, CA, USA

ACM ISBN 978-1-4503-8954-9/21/05…$15.00
https://doi.org/10.1145/3471287.3471295

1 INTRODUCTION

Big-data applications require the processing of large data sets timely and effectively. In many applications, the data are generated by devices such as sensors around the edge of the network. Edge computing is a distributed computing environment in which edge devices have substantial computing power. It can provide a solution for big-data processing by distributing the computing tasks to the edge devices that are close to the data.

Edge computing will bring great opportunities and impact to our time. Fig. 1 depicts four application fields of edge computing: home, traffic, health and city. A smart house usually contains thermometers, sensors, actuators, etc. Smart actuators can schedule temperature and humidity automatically. In smart traffic, edge devices can help analyze the traffic flow and plan routes. In smart health, edge devices can monitor the patient condition and notify the nurses when needed. Many smart objects exist in a city: smart phones, personal laptops, smart vehicles, etc. These smart devices can all respond to computing requests, and therefore, the data processing capacity can be increased by these devices. Clever use of the increased computing power can make the whole city "smart". Edge computing was reported by the US National Intelligence Council (NIC) in 2008 as a potential civil technology [1].

The concept of edge computing originates from the idea of machine-to-machine communication. Two machines on the Internet can communicate with each other and share useful resources. Edge computing proposes to utilize the computing resources of a large array of Internet devices, such as sensors, actuators, audio/video detectors, etc. Such Internet devices with computing capability can be called "smart" objects. The evolution made by edge computing is to integrate all sorts of smart objects. Processing Internet demands locally where the data are can reduce bandwidth consumption and network latency.

Resource allocation, data fusion and data security are three important research topics of edge computing. Much research work has been completed on these topics [39], [15], [22]. The distributed structure of edge computing can lead to a new issue of resource allocation. As shown in Fig. 2, cloud servers, edge servers and edge devices lie on different layers of an edge computing system. The cloud servers have high computing capacity but long data transfer delay. In contrast, the edge devices have low computing capacity and short transfer delay. The edge servers are at the middle ground with respect to both metrics. In such a system, resource allocation takes the form of selecting an optimal computing device for each computing task, and it should take into account both the computing capability and transfer delay.

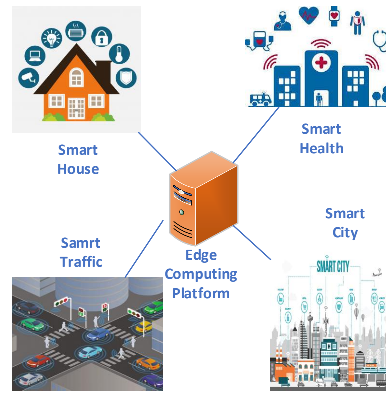

Fig. 1. Edge Computing Application Fields

The number of edge devices has drastically increased, which increases the data-processing need. Big-data processing has become a problem that edge computing must address. Furthermore, the data are generated by heterogeneous, sometimes complex, sensors. Edge computing must be effective in integrating and processing such large volume of data from a large number of heterogeneous sensors. The process to integrate and produce more accurate and consistent information is called data fusion. Data fusion removes useless information and make the raw data uniform. The effectiveness of edge computing will rely on the concise and uniform data provided by data fusion.

Malicious parties may attack an edge computing system or try to obtain the data of the users for their own benefits. Much research has been done to develop defending mechanisms [12], [23]. Edge devices, edge servers and cloud servers usually have security mechanisms installed locally, which can detect and foil attacks. In edge computing, the communication channels between different devices are additional sources of vulnerability for attacks or data leaks. A secure edge-computing system must also guard the communication channels.

We will review different approaches to deal with these three areas of problems. They are usually appropriate for specific conditions. We will explain how they can contribute to edge computing.

The rest of the paper is organized as the follows: Section 2 presents an overview of edge computing. In section 3, we describe resource-allocation techniques in edge computing. In Section 4, we discuss how to use the methods of artificial intelligence (AI) to implement data fusion and analyze the opportunities and challenges in the AI methods. In Section 5, we present existing security problems and corresponding defense mechanisms in edge computing. Emerging applications for edge computing are introduced in Section 6. The conclusion of this survey is drawn in section 7.

2 OVERVIEW OF EDGE COMPUTING

2.1 Architecture of Edge Computing

In edge computing, the edge devices are closest to the users. As such, the edge devices can provide computation services with the lowest latency. Both the edge servers and cloud servers work in an on-demand manner and provide computing results for demands from the users. In traditional cloud computing, only the cloud servers provide computing services. Edge computing has a tiered distributed structure, divided into three parts: the front-end, near-end and far-end.

2.1.1 Front-end. The edge devices are deployed at the front-end of the edge computing infrastructure. The edge devices include sensors, actuators, etc. An important type of edge devices is RFID. In RFID, track tags are attached to objects, RFID readers or devices use changing electromagnetic fields to identify objects. RFID has played a key role in actualization of the edge computing. Since the Internet users are also at the front-end, they are close to the edge devices. The Internet users can interact with the edge devices and process data without incurring obvious delay. Nonetheless, when the computing capacity of the edge devices is limited, most computation demands cannot be satisfied at the front-end environment. Then, the edge devices forward the computation requests to the near-end and far-end servers.

2.1.2 Near-end. The near-end objects include base stations, routers, edge servers, etc. In edge computing, gateways are usually deployed in the near-end environment, which control most of the traffic flows in the network. The edge servers can provide services of real-time data processing, data caching and computation offloading. Most of the data computation are executed by the near-end devices. Migrating computing tasks to the near-end devices can benefit from higher computing capability with a small increase in the communication delay compared to front-end environment.

2.1.3 Far-end. Cloud servers are usually deployed far away from the Internet users. The cloud servers in the far-end environment can provide powerful big-data processing, but the transfer time to offload data is significant compared with the front-end and near-end environments.

We next list the devices at different layers and analyze their properties. Apart from the devices, there are protocols that set the standard for the communications between different network devices.

2.2 Edge Computing Implementation

Some edge computing systems have been designed to implement the aforementioned architecture of edge computing. More generally, the design of edging computing systems mainly follows two models: the hierarchical model and the software-defined model.

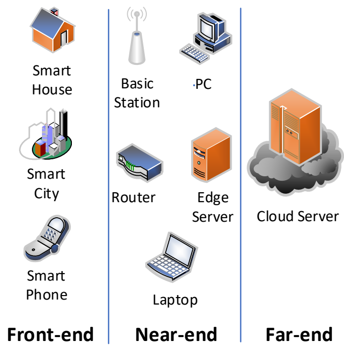

Fig. 2. A Typical Architecture of Edge Computing Networks

2.2.1 *Hierarchical Model.*

The edge servers can be deployed at different distances from the Internet users. The edge servers can be classified according to the distance to the users and their computing capacity, and these different classes result in a hierarchical edge computing architecture. A hierarchical model is suitable for describing the network structure of edge computing.

Some existing system designs fall into the hierarchical model. Kiani et al. designed a hierarchical mobile edge computing (MEC) architecture, in which the end users are provided with rapid content delivery [21]. They implemented the rapid content-delivery system with a context-aware mechanism, in which the data-processing requests are checked in the repository of the hierarchical MEC architecture. Abbas et al. explored a two-timescale approach in which the computing resources are offered in an auction-based profit-maximization manner [19]. Xi et al. proposed hierarchical-caching-based content-centric network architectures, which can support multimedia services over 5G wireless networks and satisfy various multimedia services requirements [45].

2.2.2 *Software-defined Model.*

A software defined network (SDN) attempts to centralize the network intelligence in one network component in the control plane. The control plane consists of one or more controllers, which are separated from the Internet users. SDN provides a solution that dynamically connects the computing resources with the data processing demands.

Prateek et al. proposed the LayBack architecture to facilitate communication and resource allocation among the radio access networks (RANs) devices [35]. In this LayBack architecture, the RAN functions are partitioned into function blocks that are flexibly assigned to MEC nodes. Software defined vehicular networks integrate detailed information of vehicles (e.g. location, speed, orientation) to provide services such as navigation and traffic management. For such latency-sensitive applications, edge computing can be crucial in providing timely and high-quality services [24].

2.3 Advantages of Edge Computing

Edge computing can satisfy the requirements of time-sensitive applications. Dispatching computation tasks to the edge devices can reduce latency, packet loss and bandwidth usage.

Unlike in traditional cloud computing, in edge computing, the data are either generated at or distributed to different edge devices all over the network. Since computing tasks are executed at different edge devices that are close to the data, the amount of traffic is reduced in the backbone network.

Edge computing can also balance the computing load around edge devices by carefully distributing the computation tasks of the same application to different edge computing devices. Distributing the computation tasks appropriately can also help to even out the power consumption of the devices, thus extending the system lifetime. Such task distribution can be achieved by designing and implementing a task-scheduling algorithm.

3 RESOURCE ALLOCATION

Edge computing has been considered as a promising solution to improve network performance and energy efficiency by distributing the computation tasks to the edge computing devices. On top of that, a resource allocation mechanism is also needed to effectively allocate the computations resources for different applications or tasks. In this section, we present several effective resource allocation methods.

3.1 Signal Strength Based Dynamic Resource Allocation

In mobile edge computing, a user equipment can access multiple base stations. As the user equipment moves, the base station with the highest signal strength is usually selected to serve the user. For example, the user equipment moves during the time interval from time t_1 to t_2. The user equipment can access two kinds of base stations: a macro cell (eNB) or a small cell (SCeNB). All the base stations connect with each other through the core network. As the user equipment moves, the serving base station is updated by selecting the one with the highest signal strength. A further refinement may be that the selected base station should also have enough communication capacity and computing capacity.

Oh et al. propose to select the channel with the maximum transmit power in a femto-cell network. They prove that the throughput of this network is maximized with their scheme [31]. Predictive resource allocation (PRA) techniques rely on estimation of future signal strength of the base stations. Atawia et al. use one of these techniques to ensure the quality of service (QoS) of base stations [6]. Chen et al. [11] partition the Internet users into groups based on their signal strength, and allocate resources based on the utility of the group. Rajagopal et al. use GPS to select base stations for the Internet users [32]. Their design is based on the idea that the nearest base station is likely to have the strongest signal strength.

3.2 Energy-Efficient Resource Allocation for Mobile Edge Computing

Ng et al. consider circuit power consumption, imperfect channel state information and quality of services (QoS). Synthesizing these factors, they implement an energy-efficient resource allocation algorithm in OFDMA systems [29]. By considering a Stackelberg game, Xie et al. design an energy-efficient resource allocation algorithm for heterogeneous cognitive radio networks with femtocells [42]. Meshkati et al. also rely on the game theory to implement an energy-efficient resource allocation algorithm in wireless networks [25], where edge computing can execute data processing with much less energy consumption.

Mobile edge computing provides a good platform to support augmented reality (AR) applications for mobile devices. AR mobile applications involve three parts: uplink, cloudlet for processing, and downlink. A cloudlet is a mobility-enhanced small-scale cloud datacenter that is located at the edge of the Internet. Uplink is used by the user to transmit data to the cloudlet. The cloudlet then processes the data, and the result is delivered to the user through the downlink.

For example, two users are executing a computation-intensive AR application on their mobile devices. The complex computation tasks of the AR application is transmitted to the cloudlet server for processing. The two users are executing the same AR application, they have shared inputs, outputs and computational tasks. Ali et al. designed an energy-efficient edge computing system that exploits the sharable components [4]. In their design, the system transmits input bits only from one user to the cloudlet through the uplink; multiple computation tasks are executed at the cloudlet concurrently; and the output bits are transmitted to all the users through multicast. These mechanisms can successfully reduce the energy consumption and improve the performance of the AR application.

3.3 Utility-Aware Resource Allocation for Edge Computing

The utility-aware reource allocation system has a layered architecture [43]. Smart objects exist at the bottom layer (e.g. sensors, smart gateways, and smart phones). These smart objects act as the endpoints in the edge computing platform and send computation requests to the edge computing servers. Edge servers are powerful computers placed close to the Internet users. They have higher computing capability than the edge devices. The core network provides the connections between the edge servers and the cloud big-data centers.

Most research works use a bidding strategy to implement the resource allocation [38], [17],

[20]. In the utility-aware approach, the system estimates the benefits to run the service on an edge device. The cost depends on the resources consumed by the edge device. The bids with the maximum profit have priority in obtaining the required resources. Nielsen et al. calculated the utility based on organizational citizenship behavior [30]. Aazam et al. created a resource estimation and pricing model [2]. In their model, they used the amount of services as the utility for resource allocation. Do et al. designed a resource allocation mechanism to minimize the carbon footprint [13]. Sun et al. designed an auction-based resource allocation mechanism that incorporates the prices of the edge servers [37].

Current edge computing paradigm combines these three techniques. In this paradigm, the users can be serviced by the base station with highest signal strength. Current edge computing architecture also reduces redundant data transmission. Utility-aware resource allocation can provide an appropriate resource allocation mechanism.

4 DATA FUSION

4.1 Data Fusion Architecture

A single sensor is often not enough to detect sufficient information about an object or an event. It may take a combination of multiple sensors to collect more comprehensive information. There are different architectures for the sensors to connect and fuse data. Traditional data fusion architectures are mostly centralized. However, decentralized schemes have many advantages. For instance, there is no centralized node for data processing, and therefore, no hotspot or single point of failure. Furthermore, there is no need to store all the data in a central database. Although many classical architectures have been proposed during the past decades, the most referred architecture is still the one by Joint Directors of Laboratories(JDL). Fig. 3 shows an overview of the JDL architecture. It divides the data fusion process into five steps. This hierarchical model can be used by many classification algorithms [36].

Sensor signals are the input to the fusion process. After fusing the data, the uniform output is sent to the backend computers for further processing. The main steps of data fusion process are as follows [7]:

- Level 0: Signal assessment is to obtain the refined information from raw data. The refined information is about the extracted features of the observed object.
- Level 1: Object refinement classifies the object according to the extracted features. This step can detect the identity and location of the object.
- Level 2: Situation assessment correlates the object with events by taking into account of the environment.
- Level 3: Threat assessment estimates the threats for an application based on the extracted features.
- Level 4: Process assessment optimizes the data fusion to make sure the output is best results.

There are many artificial intelligence methods associated with JDL model. These methods can successfully fuse multiple sensor data. We will introduce the general opportunities and challenges of data fusion in Section 4.2 and discuss these methods in Section 4.4.

4.2 Data Fusion Opportunities

The amount of data connected to the Internet has increased by 30% every year and will continue to increase [18]. From such a large volume of data, one needs to extract useful information and remove unnecessary or inaccurate information. To that end, the quality of the data needs to be improved. To manage the influx of data and generate reliable and accurate information, data fusion provides a good solution. Data fusion integrates available data and outputs succinct and uniform data. An important application of data fusion is multi-sensors fusion. Combining the sensor signals from multiple sensors can improve the quality of the information.

The general opportunities of data fusion using various techniques are as follows:

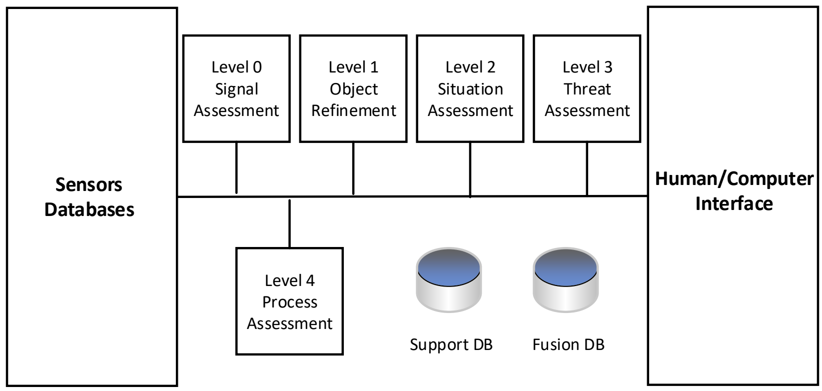

Fig. 3. The JDL Architecture [8]

- Filtering the sensor data can make data more precise, decisive and uniform.
- A sensor with higher accuracy usually consumes more power. By fusing data from multiple low-energy sensors, highly accuracy can also be achieved.
- Reducing redundant data and clustering the data based on similarity and density can improve the power efficiency.
- Big-data applications can benefit from accurate and efficient data obtained by data fusion.
- Data fusion can also hide some critical information and can be used to enhance security.

Data fusion has been widely used in the fields such as mechanical fault diagnosis, change detection, and maneuvering target tracking.

4.3 Data Fusion Challenges

Data fusion systems face several challenges, including the following:

- Heterogeneity: The sensor data from different sources are different. The heterogeneity may exist in the schema level, where the sensors express the same object in different formats. It may also exist in the instance level, where the sensors represent the same object in the real world, but the sensors have different properties. To address the heterogeneity at the schema level, we can regulate the data format using schema mapping and matching. For the heterogeneity at the instance level, we can use object matching.
- Conflicting Data: The data from different sources may conflict with each other. This is mainly caused by the incomplete sensor data, erroneous sensor data, etc. Conflicting data can lead to wrong decision-making.
- Dynamic Process: Data fusion is an iterative process. The system fuses and estimates the input data dynamically.

4.4 Artificial Intelligence Methods for Data Fusion

4.4.1 Nearest Neighbor. Nearest Neighbor algorithm is a fast and flexible method to associate data, where any point is estimated based on the K nearest points. The computation cost of such a method is very low. However, if the data set is very dense, the estimation results of many points have a high probability to be the same, which can lead to erroneous decisions.

4.4.2 Joint Probabilistic Data Association. Joint Probabilistic Data Association (JPDA) is an extension of PDA. Probabilistic Data Association (PDA) is a track focusing on a single object. Many posterior measurements are performed on the same object. The final value is achieved by averaging on the iterative measurements.

JPDA focuses on multiple targets tracking. There are multiple targets appearing in one measurement. Since the measurement index i has a probability generated by target j, some research creates a joint data association space/matrix based on the relationship between measurements and targets [33]. We associate the measurements with the states of targets through joint data association space. According to the measurements, we can update the states of multiple targets.

JPDA usually designates a valid region and only considers the results falling into the valid region. There also exist some problems for JPDA. If a new target case appears, it will just be absorbed into the old tracks instead of creating a new track. Another problem is that, one change of a single target case can trigger updating of all the surrounding tracks. Even if a track is initiated by an erroneous information, it may influence all the other tracks.

Although some problems exist, there is a convenient and efficient solution to solve it. Rezatofighi et al. narrow the surrounding range around a certain track [33]. The reduced size of a group influences less tracks, which can make the whole system more efficient and accurate.

4.4.3 Artificial Neural Networks. An artificial neural network (ANN) is a collection of units called "neurons". It is a computing system inspired by the biological neural neurons in animal brains. Artificial neural networks can make very accurate decision even based on complex and imprecise input data. Artificial neural network can extract meaningful features from complex datasets for data classification. Building an artificial neural network is an adaptive self-learning and self-organization process. It has a broad error tolerance bound. Artificial neural networks is very beneficial to data fusion, because it provides very accurate and precise prediction based on complex training and learning process.

An important application area of data fusion by ANN is vehicle control. One of such work uses ANN to control the vehicle speed and directions according to the data fusion results of the traffic information and vehicle condition data. The vehicle control system controls the speed and directions dynamically. The accuracy of estimation of vehicle is important, because the movement of the vehicle should not only obey the traffic rules but also ensure safe drive on the road. ANN is trained both by various traffic information and vehicle conditions, which can produce a minimum average error.

However, ANN trained on multiple targets may be NP hard. To this end, we sometimes use some approximation techniques to improve the performance of ANN.

4.4.4 Fuzzy Logic. Fuzzy logic algorithms use "degrees of truth" instead of the "true or false" boolean logic to describe the data. Although more fuzzy sets can lead to more accurate estimation, the computational cost of more fuzzy sets would also be higher. Therefore, it is necessary to find an appropriate number of fuzzy sets to balance the computational burden and estimation accuracy.

Fuzzy Data Association (FDA) algorithm is very similar to JPDA. Using approximated estimation, the data set has been obviously simplified. Thus data processing tasks have also been simplified.

A popular fuzzy logic algorithm is raised, which can solve track-to-track problem. The algorithm is named fuzzy clustering mean algorithm, which is classifying tracks based on attributes of each track. This algorithm maps the track to a cluster. The centroid of the cluster has the most similar attributes to the track.

The fuzzy clustering mean algorithm (FCMA) has broad tolerance of noise. The challenges of fuzzy clustering lie in determining the attributes of clusters and their related parameters. The structure of the FCMA is critical for its performance. It is also important to reduce duplicate tracks, because duplicate tracks can influence the attributes of the cluster. FCMA can provide better association effect and accurate prediction result.

4.4.5 Kalman Filter(KF). Kalman Filter has several important application areas. One of the most import applications is vehicle navigation. Kalman Filter is actually a kind of Bayesian filter. This is the optimal estimator using linear quadratic estimation (LQE). It can produce comparatively accurate estimation over a collection of observed data, imprecise data and statistical noise.

Kalman Filter is a recursive process, which feeds the global estimation results to the local filters as their prior state for next round of estimation. In every round of estimation, Kalman Filter estimates the current system state. Once an error or deviation message has been detected, the weight of the predicted state should be reduced. More weight will be given to more accurate predictions. Kalman Filter has many rounds of iterations, during this process the output of Kalman Filter becomes optimal and accurate.

The challenge of Kalman Filter lies in setting the Kalman Filter's gain, which is weight added to the predicted states. If the Kalman Filter gain value is too high, the filter follows the current state too closely. At an extreme circumstance, a high gain can lead to a jumpy trajectory. On the other hand, low gain can reduce the effects of estimation and increase the rounds to get an accurate output.

4.4.6 Particle Filter (PF). Particle filter is a filtering method to split the data set into a grid. Particle filtering uses a set of particles to estimate the posterior distribution. Particle filtering samples a set of random particles in the state space to approximate the probability density function. This estimation can be used for stochastic process and has a tolerance of noise and imprecise data. Particle filter provides an established methodology to build a nonlinear modal, which can be used to represent complex distribution, e.g., Gaussian distribution. The weighted average of selected particles can represent the Gaussian distribution results. In particle filter, the sampled data may become diverging. Several adaptive resampling criteria have been proposed to prevent diverging. Each particle has been assigned a likelihood weight to represent the probability to be sampled. If the selected data becomes too uneven, resampling step will be used. The particles with negligible weight may be replaced by those with higher weights.

5 SECURITY

5.1 Security Problems

Edge computing can detach the computation tasks from the application by offloading the data to the edge of the network. This mechanism can raise many security related problems. Privacy protection is one of our concerning issues. This section summarizes the main security risks of edge computing and introduces some methods to implement data security [44].

The security challenges exist on different layers of edge computing. Core infrastructure, edge servers and edge devices all can encounter security challenges.

The core infrastructure includes cloud servers and management systems, which may be controlled by the third party suppliers. These third party suppliers can be semi-trusted or even-untrusted. This may cause data leakage and tampering, since the sensitive information could be accessed or stolen by adversaries. On top of that, edge devices can exchange information with edge servers through core infrastructure. If the adversary uses denial of services to attack the core infrastructure, core infrastructure may be controlled by the adversary and provide false information. This is a serious security challenge, which cannot be ignored.

Edge servers are deployed at the edge of Internet. In this case, many adversaries can access the edge servers. If the adversary has controlled the edge server, they can steal and tamper the data. Moreover, the adversary can direct the information flow into his own database. Edge devices are distributed in the edge environment at different layers, such that any compromised edge device can disrupt the edge servers and gain control of the servers.

5.2 Data Security Protection Mechanism

This subsection presents several data security protection mechanisms which can be used to prevent attacks from malicious adversaries.

5.2.1 Data Confidentiality. According to the description of the reasons for security challenges. The security problems are caused mainly because the data in the edge computing are controlled by the edge servers and owned by the edge devices. These outsourced data can be accessed, tampered and lost easily. Data confidentiality scheme can provide a good solution for this problem. Data confidentiality is an encryption technique, which encrypts the outsourced data and uploads to the edge servers by the data producer. When the data users require the outsourced data, they decrypt the data from the edge servers. Current encryption techniques such as identity-based encryption [9] and attribute-based encryption [41] have raised the efficiency and operability of data confidentiality scheme.

5.2.2 Secure Data Computation. Secure data computation is an issue about searching the secure data in the edge servers. The data from Internet users are outsourced to edge servers with ciphertext form. The secure data search is a big challenge since the user has to search based on encrypted data files. Current techniques have been developed to utilize the security ranked keywords [5] and attribute-based keywords [40] to execute search jobs. These methods have proven to be accurate and efficient.

5.2.3 Authentication. Since the sensitive data are outsourced to the edge servers in edge computing, it is quite possible for the adversaries to access the data. In this case, we need to deploy some authentication enforcement approaches to deny visiting of adversaries. This mechanism can definitely protect users against existing security and privacy issues.

5.2.4 Privacy Preserving. Privacy is one of the major challenges in edge computing. Since the edge servers or core infrastructure can be accessed and abused by edge devices if they hold any certain privileges for egoistic purpose. The leakage of private information can lead to very serious results. As such, we need to propose several items to protect private information. Firstly, edge servers can take encryption techniques to store the data. Secondly, the users need to protect their identity information in the dynamic and distributed computing environment. Finally, since the users usually make use of same edge servers repeatedly, the location information is predictable. The edge server should hide the location information of users.

6 EDGE COMPUTING APPLICATIONS

An influx of data fusion applications have emerged in recent years. Recent research and technical development have put edge computing into more future-oriented application domains such as smart cities and autonomous vehicles. We will focus on several specific application areas where edge computing makes clear contributionts.

6.1 Emerging Edge Computing Applications

6.1.1 Traffic Monitoring. Edge computing is increasingly used in vehicular safety systems. A single sensor cannot cover the whole field of view needed for safety. The fusion of information from multiple sensors can broaden the field of view around the vehicle. In the fusion model, object (vehicles in our case) refinement need to be executed first. The tracks from different sensors should be aligned in space and time to make sure the captured tracks correspond to the same object. Then predict the path and detect the maneuver of the driver based on data fusion results. Vehicular active safety system is beneficial in Adaptive Cruise Control (ACC), lane change, intersection safety and so on.

6.1.2 Geodata Processing. Geodata is used in many branches of industry and academic fields. A geographic information system (GIS) is a framework for collecting, managing, and analyzing geodata. A frequent problem of geo information processing is to combine diverse data sets into a unified one, which is more consistent, accurate and useful. Data fusion, which extracts the geodata, which most matches the surface conditions provides an effective solution to this problem. By assigning attributes to the data points in the fused set using interpolation and amalgamating the extracted data features into a newly created dataset, the generated dataset is more easily identified by a GIS.

6.1.3 *Wearable Sensors Design.*

Wearable sensors are becoming pervasive in our daily life to help satisfy healthcare needs. The richness of data from wearable sensors requires uniform, accurate and relevant output. Data fusion technique is a good method to process multidimensional information. Combining medical data and daily activity data can analyze the physical conditions of a person [26].

6.2 Futuristic Edge Computing Applications

6.2.1 *Smart Cities.*

Nowadays, people are moving from rural to urban areas to search a better quality of life. This puts city resources in crisis such as water, electricity power, air, transportation, etc. There exists a necessity to manage and predict the consumption of limited resources. The prospect of future city lies in building a smart society or smart city, which is digitalized, sustainable and knowledge-managed. In edge computing, resource allocation, data fusion and security can help management and processing of data, which can increase the accuracy and form a solid base for the edge computing infrastructure.

A smart city can involve various areas of our daily life such as power distribution, resource scheduling, traffic control, health precaution, etc. In Smart City, different sensors can record different attributes of the same event. To this end, they can depict the action, identification and location of a specific object. Multi sensors are used to track objects. A method to convert angular and linear position to digital information is proposed by combing potentiometer, multiposition switch replacement, shaft encoder, and length measurement [10]. Whereas in [34], the paper designs a smart grid control system, which helps integrate distributed generation in the power distribution networks. It mainly uses a mechanism to analyze and evaluate the effects of power quality. Based on the results of quality analysis, the defected power would be isolated and the benign ones should be integrated. In [27], it suggests expressing the smart city using multi-dimensional components. The paper aligns the social infrastructure to three main dimensions (technology, people, and institutions). This can help govern and manage the infrastructures in a smart city. The smart devices like smart mobile phones and smart watches have been used more often in our daily life. The application of these devices need to access many data, which requires increasing data fusion processing. This makes the smart objects more intelligent. The hardware of smartphones consist of central processing unit (CPU), liquid-crystal display, HD voice, light and proximity sensors, and battery etc. These components provide better context awareness. Some research have been focused on data fusion using the mobile devices for human activity recognition. We need to use resource allocation technique to schedule computational resources to service different objects. Current technology is still not enough to fulfill the potentials of smart cities. There exist challenges to transform social environment into smart cities.

6.2.2 *Autonomous Vehicles.*

An autonomous vehicle is self-driving by sensing environment with little human participation. The autonomous vehicle predicts safe driving distance and plans path referring to several resources such as: GPS data, RGB camera, LIDAR signal and sensor data etc. Although it is complex to combine multiple sensors, the output can be accurate by data fusion.

A randomized path planning architecture is proposed for planning the path of kinematic and dynamic vehicles [14]. It is inspired by recent efforts using randomized algorithms. The new planning architecture can remain convergence properties. A trajectory tracking technique dealing with parametric modeling uncertainty is designed in [3]. It shows how adaptive switching supervisory control together with nonlinear Lyapunov-based tracking control can limit the tracking errors to a small range. Whereas in [16], the car is deemed as a moving sensor, absorbing information from all around environment. Based on the information, the vehicle cloud helps make the decision about the path and destinations for the customers. The internet-connected vehicles would keep information updated and guided by the vehicular cloud. In [28], some path curves are modified by using innovative polynomials such as polar polynomials and Cartesian polynomials. This modification successfully improves the tracking accuracy and avoids discontinuities existing in the traditional path. Edge computing should assign edge devices to complete the computational requests of the autonomous vehicles. When executing computing at the edge device, personal information should also be protected. The self-driving system incorporates all three techniques we introduced above.

Considering many self-driving vehicle accidents such as Google, Tesla and Uber, the technique of autonomous vehicle is still immature. We can connect autonomous vehicles to the infrastructure of edge computing, which can share data between the intelligent vehicles.

7 CONCLUSION

In this paper, we review the development status of resource allocation, data fusion and data security. We explicitly introduce the implementation mechanisms of these three dimensions and show how they can contribute to the edge computing. We also discusses the emerging and prospective application areas of edge computing.

ACKNOWLEDGMENTS

This work is funded by National Science Foundation grant CSR-1909077.

REFERENCES

[1] 2008. . Disruptive Civil Technologies: Six Technologies With Potential Impacts on US Interests Out to 2025. SRI Consulting Business Intelligence. The National Intelligence Council, National Intelligence Council, and SRI Consulting Business Intelligence. p. 48.

[2] M. Aazam and E. Huh. 2015. Fog Computing Micro Datacenter Based Dynamic Resource Estimation and Pricing Model for IoT. (2015), 687–694.

[3] A. Pedro Aguiar and JoAo P. Hespanha. 2007. Trajectory-Tracking and Path-Following of Underactuated Autonomous Vehicles With Parametric Modeling Uncertainty. *IEEE Trans. Automat. Control* 52 (2007), 1362–1379.

[4] Ali Al-Shuwaili and Osvaldo Simeone. 2017. Energy-Efficient Resource Allocation for Mobile Edge Computing-Based Augmented Reality Applications. *IEEE WIRELESS COMMUNICATIONS LETTERS* 6, 3 (2017).

[5] Mahmoud Ammar, Giovanni Russello, and Bruno Crispo. [n.d.]. Internet of Things: A survey on the security of IoT frameworks. *Journal of Information Security and Applications 38(2018)* ([n. d.]), 8–27.

[6] R. Atawia, H. Abou-zeid, H. S. Hassanein, and A. Noureldin. 2016. Joint Chance-Constrained Predictive Resource Allocation for Energy-Efficient Video Streaming. *IEEE Journal on Selected Areas in Communications* 34, 5 (2016), 1389–1404.

[7] Siwar Ben Ayed, Hanene Trichili, and Adel M. Alimi. [n.d.]. Data fusion architectures: A survey and comparison. *2015 15th International Conference on Intelligent Systems Design and Applications (ISDA)* ([n. d.]), 277–282.

[8] Erik P. Blasch and Susan Plano. 2002. JDL level 5 fusion model: user refinement issues and applications in group tracking. 4729 (2002), 270 – 279. https://doi.org/10.1117/12.477612

[9] Dan Boneh and Matthew Franklin. 2003. Identity-Based Encryption from the Weil Pairing. *SIAM J. Comput.* 32 (2003), 586 – 615.

[10] Marco Brandestini. [n.d.]. Absolute digital position encoder with multiple sensors per track.

[11] J. Chen, M. Chiang, J. Erman, G. Li, K. K. Ramakrishnan, and R. K. Sinha. 2015. Fair and optimal resource allocation for LTE multicast (eMBMS): Group partitioning and dynamics. (2015), 1266–1274.

[12] Wenxiu Ding, Xuyang Jing, Zheng Yan, and Laurence T.Yang. [n.d.]. A survey on data fusion in internet of things: Towards secure and privacy-preserving fusion. *Inf. Fusion 2019* 51 ([n. d.]), 129–144.

[13] C. T. Do, N. H. Tran, Chuan Pham, M. G. R. Alam, Jae Hyeok Son, and C. S. Hong. 2015. A proximal algorithm for joint resource allocation and minimizing carbon footprint in geo-distributed fog computing. (2015), 324–329.

[14] Emilio Frazzoli, Munther A. Dahleh, and Eric Feron. 2002. Real-Time Motion Planning for Agile Autonomous Vehicles. *AIAA J. Guid. Control* 25, 1 (2002), 116–129.

[15] K. Gai, K. Xu, Z. Lu, M. Qiu, and L. Zhu. 2019. Fusion of Cognitive Wireless Networks and Edge Computing. *IEEE Wireless Communications* 26, 3 (2019), 69–75.

[16] Mario Gerla, Eun-Kyu Lee, Giovanni Pau, and Uichin Lee. 2014. Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds. *2014 IEEE World Forum on Internet of Things (WF-IoT)* (2014), 241–246.

[17] Guocong Song and Ye Li. 2005. Utility-based resource allocation and scheduling in OFDM-based wireless broadband networks. *IEEE Communications Magazine* 43, 12 (2005), 127–134.

[18] Simon Kemp. Online. Digital trends 2019: Every single stat you need to know about the internet.

[19] Abbas Kiani and Nirwan Ansari. 2017. Toward Hierarchical Mobile Edge Computing: An Auction-Based Profit Maximization Approach. *IEEE INTERNET OF THINGS JOURNAL* 4, 6 (DECEMBER 2017).

[20] W. Kuo and W. Liao. 2007. Utility-based resource allocation in wireless networks. *IEEE Transactions on Wireless Communications* 6, 10 (2007), 3600–3606.

[21] Juyong Lee and Jihoon Lee. 2018. Hierarchical Mobile Edge Computing Architecture Based on Context Awareness. *Applied Sciences* (2018), 1160.

[22] H. Liu, Y. Zhang, and T. Yang. 2018. Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing. *IEEE Network* 32, 3 (2018), 78–83.

[23] H. Liu, Y. Zhang, and T. Yang. 2018. Blockchain-Enabled Security in Electric Vehicles Cloud and Edge Computing. *IEEE Network* 32, 3 (2018), 78–83.

[24] Jianqi Liu, Jiafu Wan, Bi Zeng, Qinruo Wang, Houbing Song, and Meikang Qiu. 2017. A Scalable and Quick-Response Software Defined Vehicular Network Assisted by Mobile Edge Computing. *IEEE Communications Magazine* 55 (July 2017), 94–100.

[25] F. Meshkati, H. V. Poor, and S. C. Schwartz. 2007. Energy-Efficient Resource Allocation in Wireless Networks. *IEEE Signal Processing Magazine* 24, 3 (2007), 58–68.

[26] Muhammad Muzammal, Romana Talat, Ali Hassan Sodhro, and Sandeep Pirbhulal. 2020. A multi-sensor data fusion enabled ensemble approach for medical data from body sensor networks. *IEEE Signal Processing Magazine* 53 (2020), 155–164.

[27] Taewoo Nam and Theresa A. Pardo. [n.d.]. Conceptualizing Smart City with Dimensions of Technology, People, and Institutions. *In Proceedings of the 12th annual international digital government research conference on digital government innovation in challenging times* ([n. d.]), 282–291.

[28] W. Nelson. 1989. Continuous-curvature paths for autonomous vehicles. *IEEE Intemational Conference on Robotics and Autoniation* (1989), 1260–1264.

[29] D. W. K. Ng, E. S. Lo, and R. Schober. 2012. Energy-Efficient Resource Allocation in OFDMA Systems with Large Numbers of Base Station Antennas. *IEEE Transactions on Wireless Communications* 11, 9 (2012), 3292–3304.

[30] Tjai M. Nielsen, Daniel G. Bachrach, Eric Sundstrom, and Terry R. Halfhill. 2012. Utility of OCB: Organizational Citizenship Behavior and Group Performance in a Resource Allocation Framework. *Journal of Management* 38 (March 2012), 668–694.

[31] D. Oh and Y. Lee. 2012. Cognitive radio based resource allocation in femto-cells. *Journal of Communications and Networks* 14, 3 (2012), 252–256.

[32] S. Rajagopal, N. Srinivasan, R. B. Narayan, and X. B. C. Petit. 2002. GPS based predictive resource allocation in cellular networks. (2002), 229–234.

[33] Seyed Hamid Rezatofighi, Anton Milan, Zhen Zhang, Qinfeng Shi, Anthony Dick, and Ian Reid. 2015. Joint Probabilistic Data Association Revisited. (December 2015).

[34] SalvadorRuiz Romero, AntonioColmenar Santos, FranciscoMur Perez, and AfricaLopez Rey. [n.d.]. Integration of distributed generation in the power distribution network: The need for smart grid control systems, communication and equipment for a smart city Use cases. *Renew Sustain Energy Rev,38(2014)* ([n. d.]), 223–234.

[35] Prateek Shantharama, Akhilesh S. Thyagaturu, Nurullah Karakoc, Lorenzo Ferrari, Martin Reisslein, and Anna Scaglione. 2018. LayBack: SDN Management of Multi-Access Edge Computing (MEC) for Network Access Services and Radio Resource Sharing. *IEEE Access* 6 (October 2018), 57545 – 57561.

[36] D. Smith and S. Singh. 2006. Approaches to Multisensor Data Fusion in Target Tracking: A Survey. *IEEE Transactions on Knowledge and Data Engineering* 18 (Dec. 2006), 1696–1710.

[37] W. Sun, J. Liu, Y. Yue, and H. Zhang. 2018. Double Auction-Based Resource Allocation for Mobile Edge Computing in Industrial Internet of Things. *IEEE Transactions on Industrial Informatics* 14, 10 (2018), 4692–4701.

[38] G. Tesauro, R. Das, W. E. Walsh, and J. O. Kephart. 2005. Utility-Function-Driven Resource Allocation in Autonomic Systems. (2005), 342–343.

[39] T. X. Tran and D. Pompili. 2019. Joint Task Offloading and Resource Allocation for Multi-Server Mobile-Edge Computing Networks. *IEEE Transactions on Vehicular Technology* 68, 1 (2019), 856–868.

[40] Changji Wang, Wentao Li, Yuan Li, and Xilei Xu. 2013. A Ciphertext-Policy Attribute-Based Encryption Scheme Supporting Keyword Search Function. *International Symposium on Cyberspace Safety and Security* (2013), 377 – 386.

[41] Shulan Wang, Junwei Zhou, Joseph K. Liu, Jianping Yu, Jianyong Chen, and Weixin Xie. 2016. An Efficient File Hierarchy Attribute-Based Encryption Scheme in Cloud Computing. *IEEE Transactions on Information Forensics and Security* 11 (June 2016), 1265 – 1277.

[42] R. Xie, F. R. Yu, H. Ji, and Y. Li. 2012. Energy-Efficient Resource Allocation for Heterogeneous Cognitive Radio Networks with Femtocells. *IEEE Transactions on Wireless Communications* 11, 11 (2012), 3910–3920.

[43] Jinlai Xu, Balaji Palanisamy, Heiko Ludwig, and Qingyang Wang. 2017. Zenith: Utility-aware Resource Allocation for Edge Computing. *2017 IEEE International Conference on Edge Computing (EDGE)* (June 2017).

[44] Jiale Zhang, Bing Chen, Yanchao Zhao, Xiang Cheng, and Feng Hu. 2018. Data Security and Privacy-Preserving in Edge Computing Paradigm: Survey and Open Issues. *IEEE Access* 6 (March 2018), 18209 – 18237.

[45] Xi Zhang and Qixuan Zhu. 2018. Hierarchical Caching for Statistical QoS Guaranteed Multimedia Transmissions over 5G Edge Computing Mobile Wireless Networks. *IEEE Wireless Communications* 25 (JUNE 2018), 12–20.