A Non-Intrusive Elderly Home Monitoring System

Le Fang, Yu Wu, Chuan Wu, Yizhou Yu

***Abstract*—Home anomaly monitoring is crucial for the elderly who live alone. A number of IoT-based home monitoring systems have been available, but most rely on privacy-intrusive cameras. With more and more concerns on privacy and security of human data, anomaly detection based on non-intrusive IoT devices becomes more desirable. Considering the elderly consumers, a low-cost system with good detection accuracy is further critical for the system's acceptability by elderly users. We propose a smart home monitoring system for living-alone senior citizens, relying on carefully designed, low-cost infrared sensor devices, as well as a cloud-based data processing and anomaly detection platform. Our PIR sensor device is effective in continuous monitoring of motion data in a user's apartment, and an open-hardware software platform is devised to support sensors manufactured by various vendors in the IoT system, all for cost reduction purpose. For privacy preservation, we encrypt collected data and store data indices in a blockchain system, to achieve efficient data access control and auditing. For motion anomaly detection, we propose a simple but effective environment adaptation method to work with the one-class Support Vector Machine (OCSVM) method. Experiments driven by real-world traces show good reliability, accuracy and efficiency of our system.**

***Index Terms*—Home monitoring, IoT, blockchain, anomaly detection.**

I. Introduction

OVER the past few decades, the aging population has rapidly increased globally and is expected to exceed 2 billion by 2050 [1].In developing and developed countries, majority of the young generation work in big cities and/or live quite far away from their elderly parents, with a very busy work or life schedule. Most older people owning their own homes would like to live in their own places, instead of in the elderly care homes, as long as they can still handle their basic daily life [2] [3]. Technology, represented by IoT systems [4]–[8], can be an enabler to promote aging in place, *i.e.*, live in one's own home and community, safely and independently [9].

Though living-alone older people can handle their basic daily living, emergency situations do happen from time to time: an old person may fall accidentally and cannot get up by him/herself, pass out due to some unexpected medical situation, or undergo some mental disturbance. For timely detection and reaction to unexpected emergencies, monitoring of elderly persons living alone has emerging as an area of significant potential in the elderly care domain.

Though using cameras for monitoring may be most effective, few elderly people would like their daily life to be under full surveillance, even if it is only their own child who is checking out the cameras. In addition, camera-based monitoring is typically limited to a few areas in a home, mainly due to the cost of camera installation. In comparison, non-intrusive monitoring using cost-effective sensors (such as light sensors, contact sensors and motion sensors) has been proposed [10] [11]. Prices for sensors also differ in a quite wide range depending on the accuracy/sensitivity and data transmission capabilities (with Bluetooth, WiFi and/or cellular connectivity). Few older persons would be willing to install an expensive smart home IoT system, with high-end sensors deployed. Therefore, we focus on developing a cost-effective IoT home monitoring system, while relying on efficient and reliable approaches and methodologies designed and deployed in our cloud data analytics platform for accurate behavior monitoring and abnormal event detection.

Our main contributions in this paper are summarized as follows:

- We investigate problems of existing home monitoring systems and carefully design a PIR (pyroelectric infrared sensor) sensing device, which is cost-efficient while effective in continuous motion monitoring in a user's apartment.
- We devise an open-hardware software platform to support sensors manufactured by various vendors in our IoT system, for further cost reduction.
- For data security, we encrypt collected data and store data indices in a blockchain system, to achieve efficient data access control and auditing.
- For cloud-based motion anomaly detection, we propose a simple but effective environment adaptation method to work with the one-class SVM method.
- We implement a prototype of our system and carry out real-world experiments. Evaluation results show good reliability, accuracy and efficiency of our system.

The rest of this paper is organized as follows. In Section II, we introduce our motivation and related work. In Section III, we describe in detail the architecture of the monitoring system. In Section IV, we analyze the design of dynamic anomaly detection. We evaluate the system performance in Section V. Finally we conclude our work in Section VI.

L. Fang, C. Wu, Y. Yu are with the Department of Computer Science, The University of Hong Kong, Hong Kong. C. Wu is also with PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen, China (email: lefang@connect.hku.hk, cwu@cs.hku.hk, yzyu@cs.hku.hk)

Y. Wu is with Southern University of Science and Technology, China and PCL Research Center of Networks and Communications, Peng Cheng Laboratory, Shenzhen, China (email: wuy7@sustech.edu.cn).

II. Motivation and Related Work

To achieve a cost-efficient high-accuracy motion anomaly detection system, we focus on three key aspects.

A. Cost-effective IoT Platform

There are several types of systems for motion monitoring: camera-based, wearable device-based, and environmental sensor-based.

Existing commercial home monitoring systems, *e.g.*, ADT Pulse [12], Vivint Smart Home [13], SimpliSafe home security system [14], Wink Lookout [15], Abode Home Security Starter Kit [16] and Xiaomi [17], mostly adopt camera-like devices. Furthermore, they are usually developed as closed ecosystems, and consumers might become "locked in", prohibiting opportunities of using sensors/products from different vendors in the system to suit their best needs (in terms of affordability and functionality).

Tyndall-DMS-Mote [18] is a wireless sensor device that monitors user vital signs within and outside their home; the user needs to carry the device round the wrist for data collection, as well as a smartphone for local data processing. Mitsufuji established a sub-brand "Hamon" [19] manufacturing smart clothing by putting IoT devices to the fabric; the devices continuously monitor data including breath, heart rate, humidity, activities, *etc.* Subasi *et al.* [20] presented an IoT system based on smart mobile devices and wearable body sensors for elderly activity monitoring. Based on the technology of Global Positioning System (GPS), Shende *et al.* [21] equipped the belt with a fall detector for activity monitoring and fall detection. With such a solution, users need to wear devices all the time for monitoring purpose.

Ramlee *et al.* [22] proposed the overall design of a Home Automation System (HAS) based on wirelessly connected appliances and environmental sensors, *e.g.*, humidity sensor and temperature sensor, which is designed to fulfill the needs of the elderly and disabled at home. Aiming to keep elderly citizens safe in their homes, IBM has carried out pilot projects equipping hundreds of apartments with IoT sensors (motion detectors, flush-detecting sensors, carbon dioxide and monoxide sensors), with collected data sent to IBM Watson platform for behavior analysis [23] [24] [25]. These works do not provide evaluation results.

Compared to using environmental sensors, the camera-based and wearable device-based systems can be more intrusive and costly [10] [11]. We adopt environmental PIR sensors for motion monitoring. As compared to systems like IBM's, our system is more lightweight and achieves better privacy preservation.

In addition, the development of IoT industries is confined in several ways, including operating system chaos and incompatible interfaces for application development. To the best of our knowledge, there is no standard software platform that can readily work with sensors from different vendors. The integration of different sensors inevitably adds to development efforts and the cost. To address this issue, we have developed an open-hardware software platform, which supports devices manufactured from different vendors in a unified manner.

B. Data Access Control

One of the most important challenges for IoT deployment is the privacy issue, as sensor devices are collecting data from human users all the time. Those data are sensitive, and who can access what portion of the data matters. For instance, some data should be exclusively restricted to the elderly person and some family members only, while some can be shared/viewed by other caregivers and/or doctors. The data storage system should desirably be able to trace the access activities too, to enforce auditing of data usage.

Blockchain, with the unique features of decentralization, traceability and cryptography, is an ideal technology to enable management of a large number of devices, and multi-levels of privacy and data security. Blockchain is a list of blocks (as shown in Fig. 1), where each block encapsulates transactions labeled with timestamps, and blocks are linked using cryptography [26]. Blockchain is by design an open distributed ledger that can record transactions/events among parties in a verifiable, permanent and distributed way. It ensures integrity and security without authentication from a third trustful party. Thus, inherently blockchain can serve as a distributed database [27]–[31], and provide plausible solutions for IoT security.

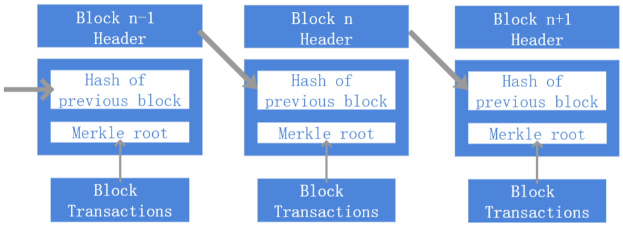


Fig. 1. Blockchain Architecture

Singh *et al.* [32] proposed a blockchain network to provide small data storage for transactions produced by IoT devices. Naveed et al. [33] proposed a blockchain-based fog computing framework for activity monitoring, which achieved multi-class cooperative categorization. Gautami *et al.* [34] stored sensory health records using the blockchain mechanism in a distributed manner. She *et al.* [35] proposed a homomorphic consortium blockchain, which was used to encrypt sensitive data of all gateway peers. Thitinan *et al.* [36] contributed to digital signature for IoT device authentication using Ethereum blockchain and implemented a one-time password for better access control in the smart home system.

Reyna *et al.* [37] pointed out challenges of integrating blockchain with IoT. As the chain grows longer, information broadcast can cause long delay for data processing. In this work, we use hybrid on-chain index storage and off-chain data storage for secure, privacy-preserving data storage. Compared with prior studies, we propose a user-based tamper-proof architecture for the storage system to fulfill the needs of privacy preservation, which provides multi-level access control based on different roles, *e.g.*, caregivers and relatives.

C. Anomaly Detection

An anomaly can be regarded as an abnormal or dangerous activity that is far from normal activities. Anomaly detection usually consists of two phases: normal daily activity recognition and abnormal behavior detection. The normal daily activity pattern is the baseline for anomaly detection. A few categories of methods have been adopted for anomaly detection.

Rule-based methods rely on domain knowledge, expert's information or specific user's habit to form detailed rules. In [38], Markov Logic Network is used to incorporate rules in both soft and hard forms. Yuan *et al.* [39] used a Fuzzy Logic System as their model to recognize activity types. A rule-based method can be more detailed and personalized for normal daily activity recognition, but requires more human efforts to formulate the rules, and can lead to more false positives since some normal behaviors may not be formulated by specific rules or ignored by the system designer.

Among *machine learning methods* for anomaly detection, one-class Support Vector Machine (OCSVM) [40] is commonly used. Abnormal behavior is rare and unexpected, and hence the normal daily activity data will be much more than abnormal data. This kind of bias in training data may degrade the performance of traditional machine learning algorithms, and OCSVM can be used to address this problem. Jakkula *et al.* [41] adopted OCSVM to build an anomaly detection system, based on collected data including date, time, sensor number, annotation from human, *etc.* Yin *et al.* [42] applied OCSVM by filtering out normal behaviour with high probability.

Another commonly used machine learning method for anomaly detection is unsupervised clustering. Hoque *et al.* [43] proposed to do clustering on every day's data in a week, separately for modeling individual activities. Lee *et al.* [44] adopted a moving window to extract features from data, and classified current activity as registered or abnormal using clustering.

Deep learning is also a choice. Erfani *et al.* [45] applied the Deep Belief Network (DBN) to extract underlying features. Lotfi *et al.* [46] used clustering to separate the normal activities and abnormal ones to produce the training dataset; then they trained a predictive model using this dataset based on the Recurrent Neural Network for binary time series prediction.

We adopt OCSVM in our system for anomaly detection, together with a simple but effective environmental adaption method. Compared with other methods, *e.g.*, support vector data description (SVDD) [47], Isolation Forest [48], we can train the OCSVM model based on the majority of normal behavior data while addressing data sample bias and avoiding the need for labeling.

III. System Design

Fig. 2 illustrates the overall design of our home monitoring system. The deployed IoT devices collect the user's motion data and then upload them to the cloud data storage system. The cloud data storage preserves the collected data and provides privacy assurance. Then the protected data would be used for data analysis, model learning, *etc.*. Specially, after detecting the anomalous condition, the emergency alarm would be sent to assigned contacts (relatives, elderly caregivers, *etc.*) instantly to avoid any secondary injuries. In the following, we detail the design of three major parts of our system. The first is on IoT deployment, including the PIR sensor model for motion tracking and the open-hardware platform for sensor compatibility. In the second part, we discuss our data storage solution based on blockchain and off-chain technologies. Next, we propose an efficient method for dynamic anomaly detection.

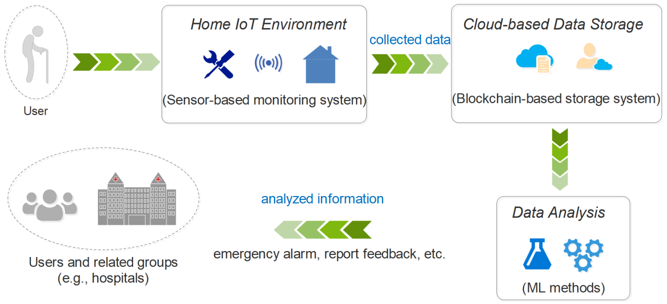


Fig. 2. Overview of the elderly home monitoring system

A. IoT Front-end

To provide reliable monitoring services for in-home older people and trigger alarms in case of any dangerous situations, varieties of sensors need to be deployed to collect and process multi-dimensional data from the caretakers, including movement, position, *etc.* These data are used to infer mood/health state and predict potential risks, *e.g.*, diseases and falls. Therefore, our system should be easy to be integrated with various sensor types, such as Pyroelectric infrared (PIR) sensors, pressure sensors, magnetic sensors and RFID readers. However, embedded systems vary widely in terms of their vendors, hardware specifications, architectures, underlying operating systems, *etc.*

Furthermore, considering the potential wide deployment of the devices in many households, the continuously collected sensor data is too overwhelming to be uploaded to the central cloud for analysis. Therefore, the IoT system should desirably provide local data processing capabilities, to save otherwise tremendous network traffic especially when the number of users grows. Instead of transmitting the live footage, the edge devices can compress or process the collected data before uploading it to the cloud. In general, apps (machine learning model, data cleaning/aggregation/compression, *etc.*) should be able to be installed flexibly in the IoT edge systems for this purpose. However, logic on IoT devices is traditionally programmed in hardware-dependent languages, which is fairly difficult to port across different platforms. Challenges arise when the elderly monitoring system involves various sensor hardware from different vendors. What's worse, future upgrades of the firmware and models on such "hard" coded platforms are simply impossible.

We design and implement an open-source middleware solution for the elderly monitoring system, which abstracts away the underlying software/hardware complexity and provides full flexibility for programming at the edge. Inspired by the success of Android and JavaME, we select Java as the language in our platform (named *Open Hardware Platform*), with more than 4000 standard class libraries out there and even more provided by active open-source developers. The runtime of our developed middleware is fully compatible with standard Java Virtual Machine (JVM), and any JVM-based technologies can be applied. In other words, with Java, enormous collections of device API are either ready or easy to expand for working with low-level protocols, *e.g.*, serial peripheral interface (SPI), inter-integrated circuit (I2C), near-field communication protocols including ZigBee, Lora, *etc.* As shown in Fig. 3, the underlying IoT operating system ("IoT OS") remains the same across different platforms and the "Open Hardware System" layer provides unified and standard Java API to the upper "apps". The "apps" refers to any application logic such as the above-mentioned machine learning, data cleaning, *etc.* Once an app is developed, our middleware solution makes it available for most different or even future products, with each optimized for specific application scenarios.

Our middleware has been ported to various existing architectures, such as Linux, Arm, etc., and is compatible with most low-level IoT operating systems, including RTThread, ThreadX, etc. Therefore, the platform can easily support various sensor devices and different communication protocols dynamically. We extracted a Java-level API for accessing generic device peripherals on embedded devices, including General Purpose Input/Output (GPIO), Inter-integrated Circuit Bus (I2C), Universal Asynchronous Receiver/Transmitter (UART) and Serial Peripheral Interface, etc. Whenever a new sensor or peripheral is to be adapted, the only job is to develop a driver app for that specific sensor. For most common types of sensors, the drivers are already either provided by the vendors or by the community. Once the driver app is installed, the IoT device can exchange data with the new sensor. A more interesting scenario is when a new sensor is detected, the driver app will be pulled automatically by the device from the cloud. Catering to resource constraint on sensor devices, our platform has a small memory footprint, requiring less than 100KB RAM, which is much lower than that of existing platforms such as Android.

Besides, we also set up extra over-the-air (OTA) servers, which can be deployed on either public cloud or private cloud. The OTA servers can communicate with and manage remote edge devices wirelessly through standard "OTA" protocols. Therefore, we can take firmware upgrades and model updates by pushing the "apps" from the cloud to the edge devices. Therefore, developers can partially update the software components while keeping the firmware unchanged. This is especially important for embedded devices where energy-saving and stability are among top concerns.

For the proof-of-concept purpose, we design and implement a 5×5 PIR sensor model (shown as Fig. 4) based on our middleware, for dynamic motion tracking in an older person's household. The core of the device consists of two Printed Circuit Boards (PCBs). The lower one Fig. 4 (a) is a simple PIR sensor array, and the upper one Fig. 4 (b) is the control unit where our middleware sits. These two PCBs are pinned together where the sensed data is transmitted. Such a loosely coupled structure is easy to expand to other sensor types. The device can cover a 5×5 square-meter area when installed in the middle of the ceiling. The device detects coarse-grained mobility information in a passive, non-intrusive manner. Fig. 5 illustrates how the PIR sensor device on the ceiling can monitor the position and trajectory of a person.

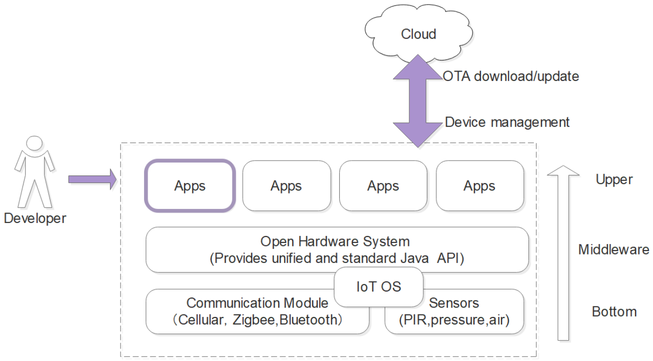


Fig. 3. Open-hardware platform

B. Blockchain-based Data Storage

Collected data, after pre-processing on the devices, will be sent to the cloud back-end for storage and analysis.

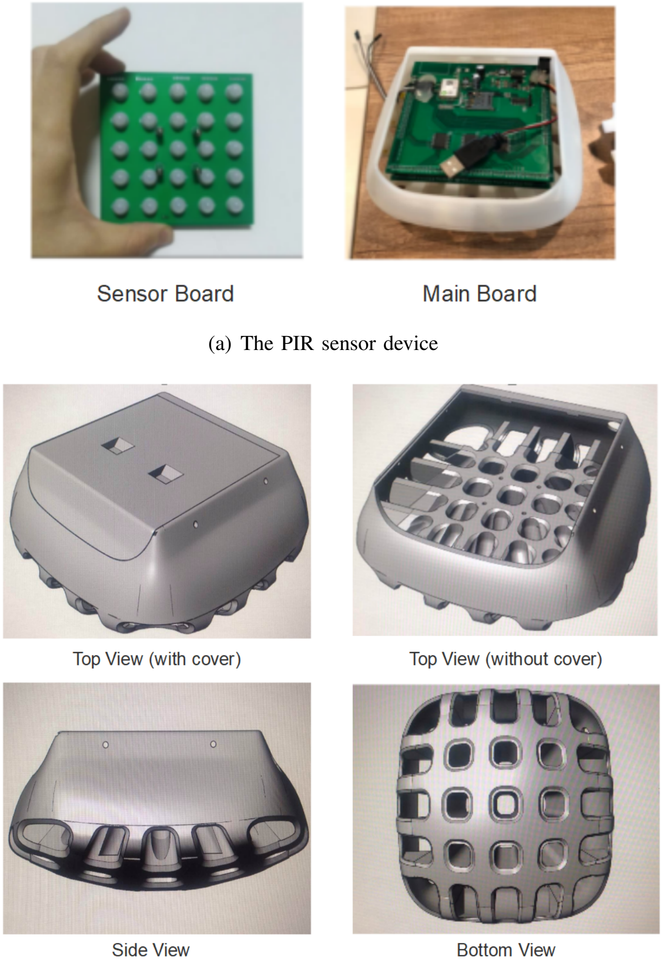
Data integrity and authenticity of personal data are critical for user acceptance of the monitoring system. Meanwhile, data sharing is important as the data should be allowed to be accessed and analyzed by third parties such as caregivers or doctors. Therefore, we design a tamper-proof storage methodology supporting secured data sharing without compromising data ownership. The fundamental functions of the storage system include:

Role-based access control. Roles are used to distinguish eligibility of users to access certain data. In our system, we associate users with different roles, including caretakers, caregivers, relatives, doctors, *etc.* Access behaviors should be strictly regulated and logged. Data are accessible only by the authorized roles.

Data Integrity and Traceability. The private data and the access logs are immutable and secured, so that any access by any role can be tracked when data leakage happens.

We adopt a blockchain-based storage system to enable multi-levels of privacy and data security. Though enabling decentralization, traceability and cryptography, blockchain is not suitable for storing large files for two reasons: 1) the block size is a hard limit on the amount of data stored on the chain; 2) its throughput is constrained by the consensus mechanisms.

Therefore, we advocate an off-chain methodology for actual data storage: the collected data is stored in the cloud in an encrypted manner off the chain, while the hash of each data block,*i.e.*, indices of data, is stored on the blockchain. The original file can not be tampered even if it is not on the chain since any change in the file would result in a



(a) The PIR sensor device

(b) The design of the device

Fig. 4. Illustration of the PIR device

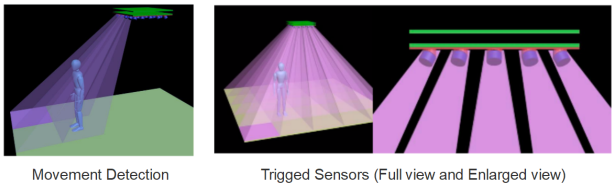


Fig. 5. Movement detection with PIR sensor device

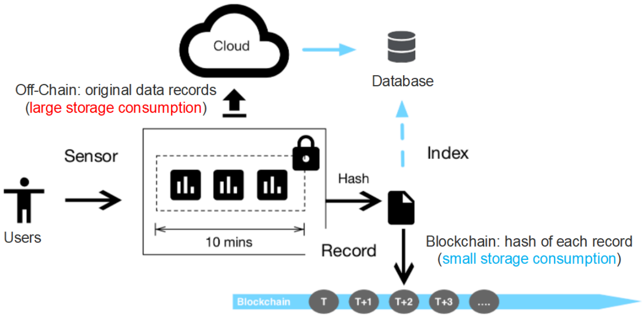

Cloud
Database
Off-Chain: original data records
(large storage consumption)
Index
Sensor
Hash
Users
10 mins
Record
Blockchain: hash of each record
(small storage consumption)
Blockchain
T
T+1
T+2
T+3
....

Fig. 6. Blockchain-based data storage

completely new hash value. Our system supports various off-chain storage types, *e.g.*, Hadoop Distributed File System (HDFS) [28], InterPlanetary File System (IPFS) [29], *etc.* IPFS is a peer-to-peer distributed file system, not relying on any central entity. Data in IPFS is managed via hashes and read/write operations are realized in a content-centric manner. Its underlying mechanisms meet the need of our design (Fig. 6) quite well, and we, therefore, implement data storage on IPFS in our prototype system. By default, the bootstrapped IPFS nodes connect to the public IPFS network which is not what we desire. Instead, we create a private network for the proof of concept, by generating a swarm key that will be referenced by all the nodes in the network.

We choose the consortium chain [49] as our underlying blockchain platform, to strike a balance between performance (throughput) and security, considering its much higher consensus efficiency as compared to the public chain [50]. As shown in Fig. 7, a consortium blockchain allows a hybrid access method. Instead of providing a fully open system by the public chain, the consensus in a consortium chain is determined by a set of dominant nodes, like family members, While the common nodes only access to read, like caregivers. Based on our hands-on experience, the most efficient method is to store a file's index/hash on the consortium chain while keeping the original file in the cloud. The footprint of a file hash is far less than that of the original file, so this solution mitigates the storage burden of the blockchain and greatly enhances the I/O throughput and transaction speed. Besides, this methodology can provide a tamper-proof architecture as any changes in the original file would result in a new hash created and stored in the consortium chain.

Our system enforces role-based access control via smart contract mechanisms. Only registered accounts can access the relevant records after the credential check by invoking the corresponding contract script. We categorize identities into different roles, *e.g.*, care-takers, family members, doctors, *etc.*, and each role can only access a specified portion of the records. If necessary, even more fine-grained access control can be realized by adding extra auditing to the contract script. In our implementation, we build the blockchain system based on Hyperledger Fabric [31] with 10 server nodes, which is a fair choice in real-life consortium blockchain system [51]. We select 3 nodes to provide the Kafka-based ordering service and the consensus efficiency is superior as compared to other counterparts, *e.g.*, pBFT. Especially, Chaincode is adopted as the "smart contract" program in Hyperledger Fabric that runs on the peers and creates transactions, which can be written to read and update the ledger state.

Fig. 6 illustrates our designed data storage system. The data stream collected from the sensors flows into our storage system, and the stream is cached and exported as file chunks at intervals. The chunk is then encrypted and uploaded to the cloud storage. Meanwhile, the chunk is hashed and the result is broadcast to the blockchain as the index of the chunk, which will be later encapsulated inside a new block. Besides chunk indexes, a block also contains metadata (filename, *etc.*) for the chunk for future query purposes and the owner's signature for later verification. We utilize the "world state" (mechanisms

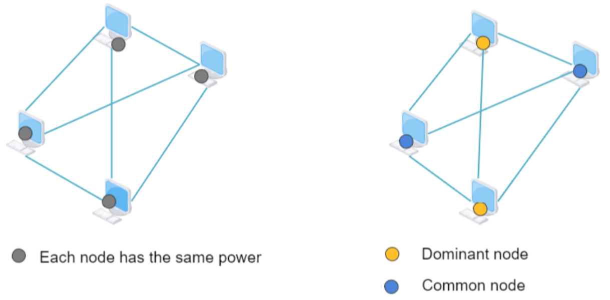


Fig. 7. Public Chain (left) vs Consortium Chain (right)

like “global variables” in Hyperledger) to track the latest chunks to expedite the query operations.

Access to the stored data happens in two phases. The indexes of the target chunks are first retrieved from the on-chain system by querying the “world state” tracker, or the chain itself if the chunks are too old for a “cache hit” in the “world state”. The retrieved index is used to access the original data from off-chain storage.

The above-mentioned write and query operations of indexes are programmed in smart contracts. For “write” operations, When indexes are to be uploaded to the chain, a `put` contract is called and this transaction will be successful after the owner’s signature is verified. The “world state” is updated once the write operation is executed on the chain. For “read” operations, a `get` contract is called with metadata of target data given, and the query indexes are returned after the owner’s approval. Our system does not enforce any specific key distribution methodologies, and any mainstream one will work.

IV. Anomaly Detection

We consider two types of anomalies in our design: still time anomaly and trajectory anomaly. Still time anomaly means that the older person stays in a position for an unexpectedly long time and trajectory anomaly takes into account the unusual movement of the user. We propose an efficient environment adaptation method for identifying indoor areas of different activity levels for abnormal still time detection and apply OCSVM for identifying abnormal trajectories.

A. Position Identification

Using our simple PIR sensor device, many sensors might get triggered simultaneously when a user is in a specific position in a room. For example, in the left figure in Fig. 5, four sensors are triggered. We identify the actual position of a user using posterior processing.

We illustrate our idea using an example case where a 5×5 PIR sensor device is installed in the center of the ceiling of a $5\times5m^2$ room. In Fig. 8, a green box represents the actual location of the person, and a grey box represents that the PIR sensor shooting at the respective grid is triggered (the person may or may not be in that grid). There are three cases:

Case 1: When the person is located in one of the green locations as in Fig. 8(a), four sensors may be triggered. For example, when the person is in the grid (2, 2), the sensors shooting at grids (1, 1), (1, 2), (2, 1) and (2, 2) are triggered.

Case 2: When the person is located in one of the green locations as in Fig. 8(b), two sensors may be triggered. For example, when the person is in the grid (2, 3), the sensors shooting at grids (2, 3) and (1, 3) are triggered.

Case 3: When the person is located in one of the green locations as in Fig. 8(c), only the sensor shooting at the particular grid will be triggered.

We can then decide the location of the person using the mappings as shown in Table I.

TABLE I
Position Mapping

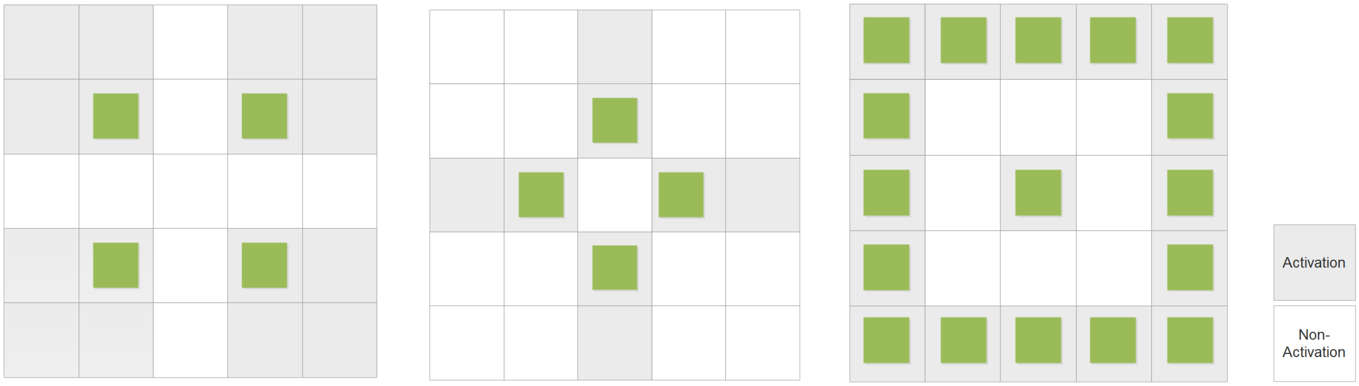
Cases	Triggered Sensors	User’s Position
Case 1	(1, 1), (1, 2), (2, 1), (2, 2)	(2, 2)
Case 1	(1, 4), (1, 5), (2, 4), (2, 5)	(2, 4)
Case 1	(4, 1), (4, 2), (5, 1), (5, 2)	(4, 2)
Case 1	(4, 4), (4, 5), (5, 4), (5, 5)	(4, 4)
Case 2	(3, 2), (2, 2)	(3, 2)
Case 2	(1, 3), (2, 3)	(2, 3)
Case 2	(3, 4), (3, 5)	(3, 4)
Case 2	(4, 3), (5, 3)	(4, 3)
Case 3	(1, 1)	(1, 1)
Case 3	(1, 2)	(1, 2)
Case 3	(1, 3)	(1, 3)
Case 3	(1, 4)	(1, 4)
Case 3	(1, 5)	(1, 5)
Case 3	(2, 1)	(2, 1)
Case 3	(2, 5)	(2, 5)
Case 3	(3, 1)	(3, 1)
Case 3	(3, 3)	(3, 3)
Case 3	(3, 5)	(3, 5)
Case 3	(4, 1)	(4, 1)
Case 3	(4, 5)	(4, 5)
Case 3	(5, 1)	(5, 1)
Case 3	(5, 2)	(5, 2)
Case 3	(5, 3)	(5, 3)
Case 3	(5, 4)	(5, 4)
Case 3	(5, 5)	(5, 5)

B. Environment Adaptation for Still Time Anomaly Detection

To facilitate anomaly detection, we divide the area under monitoring into several specific and normal divisions. The specific areas are where the user stays very long, *e.g.*, the bed area and study area; the normal areas see normal user appearances where the user usually does not stay for a long time, *e.g.*, the walking area and the entrance area.

We divide the room areas using a clustering method: we compute the average still time for each area (after position identification as the above) and cluster the results using the k means method [52]; the clusters with the longest average still times (which are much longer than the rest) correspond to specific areas and the other areas are categorized as normal areas.

Based on the area division, we propose a threshold-based method for detecting emergencies due to unexpected long still time in a position. For a specific area x_i, we set $T_{specific}[x_i]$ as the EA (Environmental Adaption) threshold for triggering an alarm. For the normal areas, we set T_{normal} as the EA threshold to issue anomaly alarms when the user has stayed in



(a) Case 1

(b) Case 2

(c) Case 3

Fig. 8. Diverse cases of sensor activation

an normal area for longer than usual. To set the EA thresholds, we record the still time in each area and use the maximal still time value in each area as the EA thresholds, $T_{specific}[x_i]$ and T_{normal}, respectively. For better performance of the method, we update the thresholds periodically, *e.g.*, once a week, twice a month.

C. OCSVM for Trajectory Anomaly Detection

We adopt OCSVM for detecting abnormal user movement trajectories. Abnormal trajectories may correspond to situations which need caregivers' attention, such as unsettled pacing due to unstable mental status, slow-than-usual walking due to unwellness. OCSVM is a special case of two-class classifiers without labeled responses, whose training focuses only on the *target class* [53]. The other class is the *outlier class*, which consists of outliers. OCSVM is trained on the samples of the target class without any outliers, and identifies a boundary around the target class. The trained classifier can be used to tell if an input sample belongs to the target class, or not (hence is an outlier, as anomaly).

Specifically, a one-class classification algorithm first maps input samples into a high dimensional feature space via a kernel function, and then finds the maximal margin hyperplane, which best separates the training data from the origin chosen from the outlier class [40].

In our case, each sample is a sequence of data points, corresponding to positions of the user (*i.e.*, each sample is a movement trajectory). Each sample corresponds to one activity of the user. In our experiments, the user moves through 3 to 7 data points (sensor triggers) according to the designed routes. We use three data sequence lengths, *e.g.*, 3, 5 and 7, for extracting features and compare their effectiveness in Section V. The default sequence length is 5 data points, as in our experiments. We adopt the Gaussian Radial Basis Function (RBF) [54] as the kernel function, which achieves a better data description by avoiding some noisy features extracted from these data sequences.

V. Evaluation

We implement a prototype of our monitoring system and deploy it in a room setting as shown in Fig. 9. A PIR device with 5 × 5 infrared sensors is installed in the center of the ceiling, which sends data to the cloud back-end through a GPRS connection of 20kbps. We implement the cloud back-end using the Spark computing framework [55] and HDFS storage system [28].

We evaluate performance of our system in four aspects.

A. IoT Deployment

To present the efficiency of the middleware platform, we evaluate its power consumption and latency from when the sensor data is collected to when the data is received by the cloud storage. During the experiment, we kept the sensors triggered and the sensor data is continuously transmitted in the form of 5 × 5 matrix with each cell indicating whether the corresponding sensor was triggered (1 - triggered, 0 - not triggered). The power consumption of our prototype sensor device is very low at less than 0.6W, due to our efficient middleware design. The latency depends on the event scheduling frequency of our middleware system, which can be configurable. We tuned it to 50 milliseconds based on our experience, which is fairly sufficient for the anomaly detection purpose.

B. Blockchain-based Storage

To prove the consortium chain in use is a feasible storage solution, we gauge the transaction throughput while storing the data and generating the hash index. Under a 10-node blockchain network, we generate random transactions as much as possible, and 107 TPS (transactions per second) is successfully appended to the chain, which is much higher than public blockchain systems such as Bitcoin and Ethereum [56], since the consensus is conducted much more efficient in consortium chain due to a much smaller number of nodes compared to the public chain system. We also conduct a stress test on our off-chain storage methodology, where we train different typical machine learning models on top of our off-chain storage system. The updated models, in terms of gradients, are saved on the blockchain after training. Table II presents the experimental results. The "Parameter Number" describes the model size. "Read" and "Write" show the upload and download time of the corresponding updated models, which is quite ideal, exhibiting practicality of our designed off-chain storage system.

TABLE II
READ/WRITE LATENCY WHEN TRAINING DIFFERENT MACHINE LEARNING MODELS ON THE CHAIN

Model	Parameter Number	Read	Write
mobileNetV2	3,538,984	155.27ms	469.72ms
mobileNet	4,253,864	153.91ms	470.09ms
NASNetMobile	5,326,716	189.37ms	489.08ms
DenseNet121	8,062,504	329.48ms	640.33ms
DenseNet169	14,307,880	891.38ms	901.10ms
DenseNet201	20,242,984	1.43s	1.24s

C. Environment Adaptation

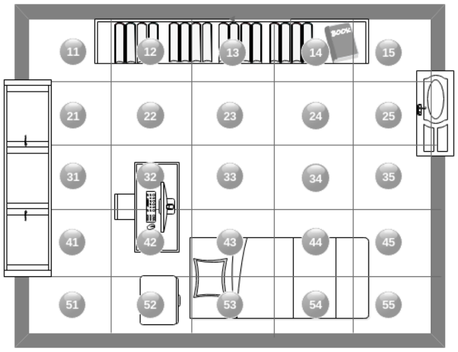
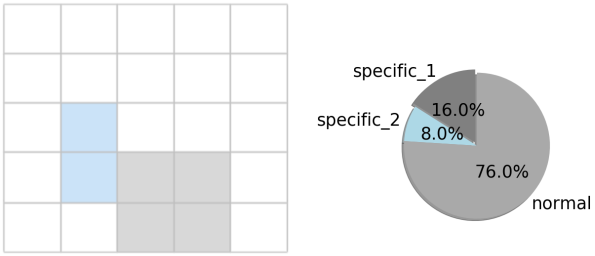


Fig. 9. Room setting for in-home monitoring

We collect our normal data set as follows. We have a user move on normal daily paths in the room in Fig. 9, emulating an older person's daily life, *e.g.*, walking to the bed, walking to the table, wandering alongside the window, walking to the bookshelf. We emulate 7-day activity monitoring with activities summarized as in Table III. We also emulate still time in the designated areas, *e.g.*, at the table, bed, *etc.* We collect 400 sample trajectories, each containing a sequence of activation time and sensors triggered along the path (default trajectory length is 5).



(a) normal(white), specific_1(grey), specific_2(blue)

(b) Percentages of specific/normal areas

Fig. 10. Area Division

Fig. 10 shows the specific/normal area division results. The percentage of specific/normal areas in terms of the entire room area is given in Fig. 10(b). The "specific" areas with the longest still time, labeled with IDs 43, 44, 54, 55 (Fig. 9), are the bed areas of the room, while the other set of "specific" areas are the study areas of the bedroom with IDs 43 and 44. The rest are "normal" areas.

We derive the maximal still time in an area as the EA threshold for triggering emergency alarms for that area. Table IV gives the results. We have $T_{specific}[x_1] = 31046.891$, $T_{specific}[x_2] = 3651.004$ and $T_{normal} = 4.977$.

D. Anomaly Detection

We further collect 100 samples with random paths as anomaly data. As abnormal situations happen rarely in reality, we set the ratio of normal samples and abnormal samples to 10 : 1 in the following experiments (400 normal samples vs 40 random anomaly samples by default), which is a common ratio used for anomaly detection in existing literature [42], [57].

We evaluate the performance of our OCSVM abnormal trajectory detection using the following metrics, where TP stands for true positive, FP is false positive, TN is true negative and FN is false negative, respectively:

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$FPR = \frac{FP}{FP + TN} \tag{2}$$

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

Here, *Recall*, also referred to as the true positive rate (TPR), is the proportion of real positive cases (anomaly) that are correctly predicted. Conversely, *Precision* denotes the proportion of predicted positive cases that are correctly real positives. *FPR* presents the proportion of real negative cases that are predicted positive [58]. We plot the Precision-Recall (PR) curve with the recall value on x-axis and the precision value on y-axis, and the Receiver Operating Characteristic (ROC) curve with the FPR value on x-axis and the TPR value on y-axis. We compute the Area under Curve (AUC) for these curves, which is the normalized area under a curve, for evaluating the performance of the classifier [59]: the larger AUC is, the better capability of distinction between classes is achieved. AUC has been approved to be an effective method for measuring the performance of classifiers with unbalanced class distribution and multiple classes [60].

We plot the PR and ROC curves in Fig. 11 and Fig. 12 when the length of data sequence per sample is 5. We compare the impact of the length of the data sequence for anomaly detection in Fig. 13: we run each experiment 10 times, and present the average AUC values as well as the maximal and minimal values in each case. The average AUC value of the PR Curve reaches the largest value (0.8694) when the sequence length is 7, whereas the value is the smallest (0.4825) if the length is 3. As the sequence gets longer, it contains more information along the trajectory, potentially capturing more characteristics of the normal activities, such that the anomaly can be distinguished more easily. The average AUC

TABLE III
7-DAY IN-HOME MONITORING

Day	Activities	Position
Day 1	Enter the room → go to bed	25 → 53, 43, 54, 44
Day 2	Get up → the window site → leave the room → enter the room → lunch break → window site → read → leave the room → enter the room → go to bed	53, 43 → 21, 31, 41 → 25 → 53, 43 → 31 → 32, 42 → 25 → 53, 43, 54, 44
Day 3	Get up → leave the room → enter the room → read → leave the room → enter the room → lunch break → read → leave the room → enter the room → go to bed	53, 43, 54, 43 → 25 → 12, 13, 14 → 32, 42 → 25 → 53, 43 → 32, 42 → 25 → 53, 43
Day 4	Get up → leave the room → enter the room → the window site → leave the room → enter the room → lunch break → read → leave the room → enter the room → go to bed	53, 43 → 25 → 21,31,41 → 25 → 53,43 → 12,13,14 → 32,42 → 25 → 53,43
Day 5	Get up → leave the room → enter the room → lunch break → read → leave the room → enter the room → go to bed	53,43 → 25 → 53,43 → 31,21,41 → 32,42 → 25 → 53,43
Day 6	Get up → leave the room → enter the room → read → lunch break → read → leave the room → enter the room → go to bed	53,43 → 25 → 42,32 → 53,43 → 31,21,41 → 32,42 → 25 → 53,43
Day 7	Get up → leave the room → enter the room → lunch break → leave the room → enter the room → read → go to bed	53,43 → 25 → 53,43 → 25 → 31,21,41 → 32,42 → 53,43

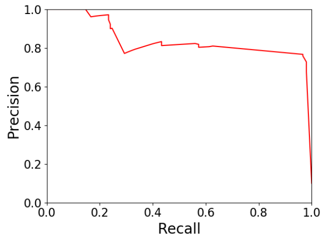


Fig. 11. PR Curve

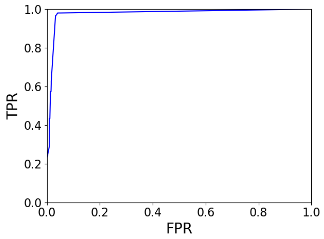


Fig. 12. ROC Curve

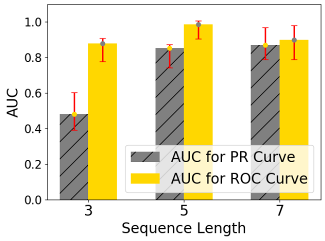


Fig. 13. AUC with different sequence lengths

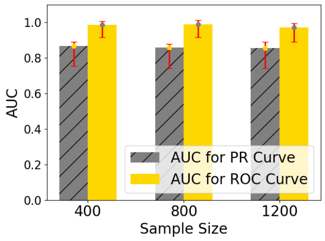


Fig. 14. AUC with different training dataset sizes

TABLE IV
STILL TIME IN DIFFERENT AREAS

Areas	Average Still Time (seconds)	Maximal Still Time (seconds)
specific_1 (53, 43, 54, 44)	15674.141	31046.891
specific_2 (32, 42)	2975.951	3651.004
normal (the rest)	3.425	4.977

of the ROC Curve achieves the largest value (0.9764) when the sequence length is 5, which avoids more false positive examples.

We next increase the number of samples used for training our OCSVM classifier and present the anomaly detection results in Fig. 14. We observe that the performance does not change much with more training samples. We summarize the results in Table V. We observe that we can obtain a better AUC of PR by increasing the sequence length. As each trajectory consists of continuous data points, a longer sequence can capture normal activities better by containing more information.

E. Comparison with Existing Systems

We compare our results with existing IoT systems which focus on activity recognition and anomaly detection [42], [61]–[69], and show the results in Table VI. These works adopted diverse methods for achieving activity recognition and anomaly detection, *e.g.*, Support Vector Machine (SVM) [70], Dynamic Range-Doppler Trajectory (DRDT) [67], Bayesian statistics [71], convolutional neural network (CNN) [72], generative adversarial network (GAN) [73], principal component analysis network (PCANet) [74], kernel nonlinear regression (KNLR) [75], *etc.*. Note that F-Measure in [61]–[63] is a metric combining both precision and recall, as follows:

$$F - Measure = \frac{2}{precision^{-1} + recall^{-1}} \tag{4}$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \tag{5}$$

Comparing with the existing sensor-based anomaly detection systems (*e.g.*, [42] [68] [64] [69]), our system provides comparable performance while replying on simpler environmental sensor devices and methods. Comparing with our system, video-based systems (*e.g.*, [65] [66]) are more expensive and hard to implement in the elderly household environment, due to severe privacy and cost concerns.

VI. CONCLUSION

This paper proposes a cost-effective, efficient smart home monitoring system for living-alone elderly people. The key components include a non-intrusive, efficient front-end sensor device and open-hardware middleware, blockchain-based data storage, and simple but effective anomaly detection methods. Our PIR device can be easily installed in the home environment and connected with extra online services via the open-hardware platform. Our cloud data storage provides secure,

TABLE V
SUMMARY OF TRAJECTORY ANOMALY DETECTION RESULTS

Sequence Length	Samples	Ave_AUC(PR)	Ave_AUC(ROC)	Max_AUC(PR)	Max_AUC(ROC)
5	**400**	**0.8531**	**0.9764**	**0.8712**	**0.9863**
3	400	0.4825	0.8789	0.6024	0.9039
7	400	0.8694	0.8995	0.9540	0.9602
5	800	0.8592	0.9679	0.8791	0.9893
5	1200	0.8571	0.9701	0.8807	0.9826

TABLE VI
PERFORMANCE COMPARISON WITH EXISTING IOT MONITORING SYSTEMS

Ref.	Type	IoT deployment	Method	Detection objectives	Performance
[61]	Activity Recognition	MEMS sensors	SVM	2 kinds of activities: atom activities (cook lunch/breakfast, eat lunch, etc.) and interactive activities (sleep, walk, leave home, etc.)	Accuracy:0.87, Precision:0.82, F_Measure :0.848
[62]	Activity Recognition	Micro-Doppler Radar	GAN	7 kinds of activities: boxing while standing in place, crawling on the ground, creeping, jumping forward, running, standing and walking	Max F-Measure : 0.7461
[63]	Activity Recognition	Acceleration and gyro sensor	CNN	Diverse transportation methods: ride a car/bike, take the bus/train/subway, run, walk and keep still	Max F-Measure : 0.82
[67]	Activity Recognition	FMCW Radar	DRDT	6 typical human motions: falling, stepping, Jumping, squatting, walking and jogging	Average accuracy : 0.919
[64]	Activity Recognition	Accelerometers and gyroscopes of mobile phones	CNN	6 types of physical activities, walk, walk-upstairs, walk-downstairs, sit, stand and lay.	Max accuracy : 0.93926
[65]	Activity Recognition	Camera	CNN	1000 object classes	Max accuracy : 0.875
[66]	Anomaly Detection	Video surveillance	PCA+CNN+SVM	Motion tracking and anomalous object detection	Max accuracy : 0.94
[42]	Anomaly Detection	Wearable Sensors	SVM+KNLR	112 abnormal traces like falling down backwards/forwards, slipping, *etc.*	AUC of ROC: 0.985
[68]	Anomaly Detection	Infrared (IR) Motion Sensors	SVDD	10 different types of abnormal conditions: falls, weakness, seizures, dead, etc.	AUC of ROC : 0.96
[69]	Anomaly Detection	RFM Wireless Sensors	Bayesian Statistics	Daily activity monitoring of 3 people: an autonomous and independent adult man, an elderly woman with Parkinson's disease and an autonomous elderly woman	AUC of ROC : 0.9455
Our work	Anomaly Detection	PIR Sensors	EA+OCSVM	Still time anomaly and trajectory anomaly	AUC of ROC: 0.9863

multi-level access control by adopting both on-blockchain and off-chain technologies. For anomaly detection, we propose an environment adaptation method for still time anomaly detection and adopt OCSVM for trajectory anomalies. We implement a prototype system and evaluate it in a real-world environment. Extensive experiments show timeliness, low cost and accuracy of our solutions.

For future work, we are planning to conduct long-term behavior studies of the elder users based on more advanced machine learning methods, to learn more valuable information such as emotional and psychological status.

ACKNOWLEDGEMENT

This work was supported in part by grants from Hong Kong RGC under the contracts HKU 17204619, 17225516, C5026-18G (CRF) and Hong Kong ITF (ITS/305/19FP). This work was also supported by NSFC under grant No.61702098 and PCL Future Greater-Bay Area Network Facilities for Large-scale Experiments and Applications (LZC0019).

REFERENCES

[1] Ageing and health. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/ageing-and-health

[2] Proportion of population aged 65 and over. [Online]. Available: https://ec.europa.eu/eurostat/tgm/graph.do?tab=graph&plugin=1&pcode=tps00028&language=en&toolbox=sort

[3] Aging in place. [Online]. Available: https://www.aginginplace.org

[4] A. Samanta and Z. Chang, "Adaptive service offloading for revenue maximization in mobile edge computing with delay-constraint," *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3864–3872, 2019.

[5] A. Samanta and J. Tang, "Dyme: Dynamic microservice scheduling in edge computing enabled iot," *IEEE Internet of Things Journal*, 2020.

[6] A. Samanta, L. Jiao, M. Mühlhäuser, and L. Wang, "Incentivizing microservices for online resource sharing in edge clouds," in *Proc. of the IEEE 39th International Conference on Distributed Computing Systems*, July 2019.

[7] A. Samanta, Y. Li, and F. Esposito, "Battle of microservices: Towards latency-optimal heuristic scheduling for edge computing," in *Proc. of the IEEE 5th Conference on Network Softwarization*, June 2019.

[8] D. Xu, A. Samanta, Y. Li, M. Ahmed, J. Li, and P. Hui, "Network coding for data delivery in caching at edge: Concept, model, and algorithms," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 10, pp. 10 066–10 080, 2019.

[9] E. Hwang, L. Cummings, A. Sixsmith, and J. Sixsmith, "Impacts of home modifications on aging-in-place," *Journal of Housing for The Elderly*, vol. 25, pp. 246–257, July 2011.

[10] S. Robben, M. Pol, and B. Krose, "Longitudinal ambient sensor monitoring for functional health assessments: a case study," in *Proc. of the*

2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing, September 2014.

[11] S. Abd El-Kader and B. Mohammad El-Basioni, "Independent living for persons with disabilities and elderly people using smart home technology," *International Journal of Application or Innovation in Engineering & Management*, vol. 3, pp. 11–28, April 2014.

[12] Adt pulse. [Online]. Available: https://www.adt.com/pulse

[13] Vivint smarthome. [Online]. Available: https://www.vivint.com

[14] Simplisafe. [Online]. Available: https://simplisafe.com

[15] Wink look out. [Online]. Available: https://www.wink.com/products/wink-lookout-smart-security-essentials/

[16] Abode home security starter kit (2018) review: This entry-level smart home hub continues to evolve. [Online]. Available: https://www.techhive.com/article/3259876/abode-review.html

[17] Xiaomi smarthome. [Online]. Available: https://xiaomi-mi.com/mi-smart-home/

[18] J. Donoghue, J. Herbert, and P. Stack, "Remote non-intrusive patient monitoring," in *Proc. of 4th International Conference on Smart Homes and Health Telematics*, June 2006.

[19] Hamon. [Online]. Available: http://www.hamon.tech/en/#products

[20] A. Subasi, M. Radhwan, R. Kurdi, and K. Khateeb, "IoT based mobile healthcare system for human activity recognition," in *Proc. of the IEEE 15th Learning and Technology Conference*, February 2018.

[21] D. Shende, S. Madrewar, and S. Dugade, "Dementia Patient Activity Monitoring and Fall Detection using IoT for Elderly," *International Journal of Trend in Scientific Research and Development*, vol. 3, pp. 363–367, June 2019.

[22] R. Ramlee, M. A. Othman, M. Leong, M. Ismail, and S. Ranjit, "Smart home system using android application," in *Proc. of the International Conference of Information and Communication Technology*, March 2013.

[23] IBM Watson IoT Platform. [Online]. Available: https://www.ibm.com/hk-en/marketplace/internet-of-things-cloud

[24] Elder care organization. [Online]. Available: https://www.ibm.com/case-studies/z976639f72075f89

[25] AI technology brings innovation to elderly care. [Online]. Available: https://www.ibm.com/blogs/client-voices/ai-technology-innovation-elderly-care/

[26] Blockchains: The great chain of being sure about things. [Online]. Available: https://www.economist.com/briefing/2015/10/31/the-great-chain-of-being-sure-about-things

[27] Storj. [Online]. Available: https://storj.io/

[28] Hadoop. [Online]. Available: http://hadoop.apache.org/

[29] IPFS. [Online]. Available: https://ipfs.io/

[30] IPLD. [Online]. Available: https://ipld.io/

[31] Hyperledger fabric. [Online]. Available: https://www.hyperledger.org/use/fabric

[32] M. Singh, A. Singh, and S. Kim, "Blockchain: A game changer for securing IoT data," in *Proc. of the IEEE 4th World Forum on Internet of Things*, February 2018.

[33] N. Islam, Y. Faheem, M. T. Ikram Ud Din, M. Guizan, and M. Khalil, "A blockchain-based fog computing framework for activity recognition as an application to e-Healthcare services," *Future Generation Computing Systems*, vol. 100, pp. 569–578, November 2019.

[34] G. Tripathi, M. Ahad, and S. Paiva, "S2hs- a blockchain based approach for smart healthcare system," *Healthcare*, p. 100391, November 2019.

[35] W. She, Z. Gu, X. Lyu, Q. Liu, T. Zhao, and W. Liu, "Homomorphic Consortium Blockchain for Smart Home System Sensitive Data Privacy Preserving," *IEEE Access*, vol. 7, pp. 62 058–62 070, May 2019.

[36] T. Tantidham and Y. N. Aung, "Emergency Service for Smart Home System Using Ethereum Blockchain: System and Architecture," in *Proc. of IEEE International Conference on Pervasive Computing and Communications Workshops*, March 2019.

[37] A. Reyna, C. Martín, J. Chen, E. Soler, and M. Díaz, "On blockchain and its integration with IoT Challenges and opportunities," *Future Generation Computing Systems*, vol. 88, pp. 173–190, November 2018.

[38] K. Gayathri, S. Elias, and B. Ravindran, "Hierarchical activity recognition for dementia care using markov logic network," *Personal and Ubiquitous Computing*, vol. 19, no. 2, pp. 271–285, 2015.

[39] B. Yuan and J. Herbert, "Context-aware hybrid reasoning framework for pervasive healthcare," *Personal and ubiquitous computing*, vol. 18, no. 4, pp. 865–881, 2014.

[40] B. H. Schölkopf, J. C. Platt, J. C. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution," *Neural Computation*, vol. 12, pp. 1443–1471, 2001.

[41] V. Jakkula and D. Cook, "Detecting anomalous sensor events in smart home data for enhancing the living experience," in *Workshops at the 25th AAAI Conference on Artificial Intelligence*, August 2011.

[42] J. Yin, Q. Yang, and J. J. Pan, "Sensor-Based Abnormal Human-Activity Detection," *IEEE Transactions on Knowledge and Data Engineering*, vol. 20, pp. 1082–1090, June 2008.

[43] E. Hoque, R. F. Dickerson, S. M. Preum, M. Hanson, A. Barth, and J. A. Stankovic, "Holmes: A comprehensive anomaly detection system for daily in-home activities," in *Proc. of the International Conference on Distributed Computing in Sensor Systems*, June 2015.

[44] M.-S. Lee, J.-G. Lim, K.-R. Park, and D.-S. Kwon, "Unsupervised clustering for abnormality detection based on the tri-axial accelerometer," in *Proc. of ICCAS-SICE International Joint Conference*, August 2009.

[45] S. M. Erfani, S. Rajasegarar, S. Karunasekera, and C. Leckie, "High-dimensional and large-scale anomaly detection using a linear one-class svm with deep learning," *Pattern Recognition*, vol. 58, pp. 121–134, 2016.

[46] A. Lotfi, C. Langensiepen, S. M. Mahmoud, and M. J. Akhlaghinia, "Smart homes for the elderly dementia sufferers: identification and prediction of abnormal behaviour," *Journal of ambient intelligence and humanized computing*, vol. 3, no. 3, pp. 205–218, 2012.

[47] D. Tax and R. Duin, "Support Vector Data Description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, January 2004.

[48] F. T. Liu, K. M. Ting, and Z.-H. Zhou, "Isolation Forest," in *Proc. of the IEEE 8th International Conference on Data Mining*, December 2008.

[49] O. Dib, K.-L. Brousmiche, A. Durand, E. Thea, and E. Ben Hamida, "Consortium blockchains: Overview, applications and challenges," *International Journal On Advances in Telecommunications*, pp. 51–64, September 2018.

[50] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, "An overview of blockchain technology: Architecture, consensus, and future trends," in *Proc. of the IEEE 6th International Congress on Big Data*, June 2017.

[51] R. v. Mölken. (2018) Blockchain across oracle : understand the details and implications of the blockchain for oracle developers and customers. [Online]. Available: http://proquest.safaribooksonline.com/?fpi=9781788474290

[52] K. Wagstaff, C. Cardie, S. Rogers, and S. Schroedl, "Constrained k-means clustering with background knowledge," in *Proc. of the 18th International Conference on Machine Learning*, January 2001.

[53] D. Tax, "One-class classification," Ph.D. dissertation, Delft University of Technology, 2001.

[54] S. Amari and S. Wu, "Improving support vector machine classifiers by modifying kernel functions," *Neural Networks*, pp. 783–789, 1999.

[55] Spark. [Online]. Available: http://spark.apache.org/

[56] Ethereum. [Online]. Available: hhttps://ethereum.org

[57] R. Kishi, P. T. Huy, K. Yamamoto, and M. Masuda, "Abnormal behavior detection using image sensing," *OKI Technical Review*, pp. 1–4, May 2019.

[58] Powers and D. Martin, "Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation," *Journal of Machine Learning Technologies*, pp. 37–63, 2011.

[59] J. Davis and M. Goadrich, "The Relationship Between Precision-Recall and ROC Curves," in *Proc. of the 23rd International Conference on Machine Learning*, June 2006.

[60] C. Ling, J. Huang, and H. Zhang, "Auc: a statistically consistent and more discriminating measure than accuracy," in *Proc. of the 18th International Joint Conference on Artificial Intelligence*, August 2003.

[61] J. H. Guo, Y. Mu, M. D. Xiong, Y. Q. Liu, and J. X. Gu, "Activity Feature Solving Based on TF-IDF for Activity Recognition in Smart Homes," *Complexity*, vol. 2019, March 2019.

[62] Y. Yang, C. P. Hou, Y. Lang, D. Guan, D. Y. Huang, and J. C. Xu, "Open-set human activity recognition based on micro-Doppler signatures," *Pattern Recognition*, vol. 85, pp. 60–69, January 2019.

[63] C. Ito, M. Shuzo, and E. Maeda, "CNN for human activity recognition on small datasets of acceleration and gyro sensors using transfer learning," in *Adjunct Proc. of the 2019 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2019 ACM International Symposium on Wearable Computers*, September 2019.

[64] S. Dhanraj, S. De, and D. Dash, "Efficient smartphone-based human activity recognition using convolutional neural network," in *Proc. of the IEEE 18th International Conference on Information Technology*, December 2019.

[65] S. Eum, C. Reale, H. Kwon, C. Bonial, and C. Voss, "Object and text-guided semantics for cnn-based activity recognition," in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2019.

[66] A. Mohan, M. Choksi, and M. A. Zaveri, “Anomaly and activity recognition using machine learning approach for video based surveillance,” in *Proc. of the IEEE 10th International Conference on Computing, Communication and Networking Technologies*, July 2019.

[67] C. Ding, H. Hong, Y. Zou, H. Chu, X. H. Zhu, F. Fioranelli, J. L. Kernec, and C. Li, “Continuous Human Motion Recognition With a Dynamic Range-Doppler Trajectory Method Based on FMCW Radar,” *IEEE Transactions on Geoscience and Remote Sensing*, vol. 57, no. 9, pp. 6821–6831, April 2019.

[68] J. H. Shin, B. Lee, and K. S. Park, “Detection of Abnormal Living Patterns for Elderly Living Alone Using Support Vector Data Description,” *IEEE Transactions on Information Technology in Biomedicine*, vol. 15, no. 3, pp. 438–448, May 2011.

[69] F. J. Ordóñez, P. de Toledo, and A. Sanchis, “Sensor-based Bayesian detection of anomalous living patterns in a home setting,” *Personal and Ubiquitous Computing*, vol. 19, no. 2, pp. 259–270, February 2015.

[70] C. Cortes and V. Vapnik, “Support-vector networks,” *Machine Learning*, pp. 273–297, 1995.

[71] A. Gelman, J. B. Carlin, H. S. Stern, D. B. Dunson, A. Vehtari, and D. B. Rubin, *Bayesian Data Analysis*. CRC Press, 2013.

[72] S. Lawrence, C. Giles, A. Tsoi, and A. Back, “Face recognition: a convolutional neural-network approach,” *IEEE Transactions on Neural Networks*, vol. 8, pp. 98–113, February 1997.

[73] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. W.-F. S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial networks,” in *Proc. of the 28th Conference on Neural Information Processing Systems*, December 2014.

[74] S. Wold, K. Esbensen, and P. Geladi, “Principal component analysis,” *Chemometrics and Intelligent Laboratory Systems*, pp. 37–52, August 1987.

[75] I. W. Tsang, J. T. Kwok, B. Mak, K. Zhang, and J. J. Pano, “Fast speaker adaption via maximum penalized likelihood kernel regression,” in *Proc. of the IEEE International Conference on Acoustics, Speech and Signal Processing*, May 2006.