Information and Communication Technology-based
Tele-Monitoring
for Elderly Care Houses

Hisato KOBAYASHI, Yoshiyuki HINO
IT Research Center
Hosei University
Tokyo, Japan
h, it@hosei.ac.jp

Ivan HO, Billy PHAM, Shoichiro WATANABE
Hosei University Research Institute, California
Burlingame, California, USA
ivan, bpham, shou@huric.org

Abstract—This paper reports the deployment of a tele-health monitoring system to be installed at an actual care house, which can monitor heart and breathing rates, and in/out of bed status of elderly residents over the Internet. Through a one-month operation, we could recognize several serious problems. This paper indicates these problems and suggests solutions to overcome them.

Keywords—health monitoring, tele-monitoring, elderly care

I. INTRODUCTION

The world population is aging. Every country, rich or poor, developed or still developing, has a population that is aging to some extent. The number of elderly people will rise to 1.4 billion in 2040, which is about twice as much as the current elderly population; to put it in perspective, 14% of people worldwide will be older than 65 years of age, according to a report by the U.S. Census Bureau [1].

With age, people are likely to have diseases and/or injuries when performing physical or daily activities. In the worst case, elderly individuals can become bedridden for the rest of their lives. In such case, a bedridden elderly needs about 8,760 h of care per year; with an average fulltime work roughly representing 1,800 h per year, a bedridden elderly would need at least five caregivers. If 10% of the population were to become bedridden, about 50% of the population would be needed to care for those 10%, which would be a very worrisome issue. Furthermore, the U.S. Census Bureau states that in the near future, the elderly population will exceed the young population, which means that there will be fewer caregivers available for the elderly.

To overcome this problem in an aging society, we need to improve the efficiency of care by using information and communication technology (ICT). In our research, we have focused on an ICT-based tele-health monitoring system for care houses. This technology essentially allows caregivers to monitor elderly, including bedridden elderly, from a remote location and provide the required care when necessary. Additionally, data collected by the system can be further analyzed for caregivers to construct accurate care plans and prevent diseases or accidents [2] [3].

First, we need to know every detail about elderly care houses and allot appropriate labor task at suitable timings.

II. OUTLINE OF DEPLOYMENT

A. Design of the Monitoring System

The recent progress of wireless communication technology makes it possible to transmit vital signs such as blood pressure, pulse, and breathing, to a cloud server for real-time analysis. Many types of such devices have been developed but most of them are only usable by wearing or attaching them to the body. Moreover, many of these devices require users to manually send data to a server. Therefore, it is almost impossible for elderly people, especially dementia patients to use them. Instead, we tried to design an unrestrained and unobtrusive system for continuous monitoring of their vital signs [4].

First, we developed a system that monitors heartbeat, breathing, and body movement while an elderly is in bed because the quality of sleep is essential for good health. Moreover, sleep deprivation and sleep disorders weaken the immune system. Especially for patients with dementia, it is thought that sleep is an important factor to maintain their cognitive functioning [5] [6]. Early detection of such sleep disorders helps patients not only maintaining their health condition but also reducing care costs. Care costs do not only refer to medication costs but also include indirect overheads such as the labor cost in care houses. In the care house where we first implemented our system, each caregiver takes care of 2–3 patients during daytime, but almost three times more patients during nighttime. Hence, it is a burden for a sole caregiver to watch and take care of such a number of elderly during night shifts. Our system releases caregivers from having to visit rooms all night long, and alternatively enables them to monitor online all the rooms simultaneously.
As a next step, we may consider increasing the variety of sensors so that we can analyze the data more accurately and send an alert to caregivers timely.

B. System Components

The following are the key components of our system (Figure 1):

1) **Sleep monitoring system:** A mat-based sleep monitor (sensor sheet) is used to collect vital signs of the elderly, such as heartbeat and breathing rates and body movement, by placing it under the mattress. The sensor sheet also detects whether an elderly is “in bed” or “out of bed,” and “awake” or “asleep.”

2) **Emergency call system:** An Arduino micro PC is used to detect emergency call signals and send information automatically to the cloud server. The emergency call button is placed on the leg of the headboard so that elderly can easily reach it without getting up.

3) **Headboard:** For this experiment, we designed a headboard where we store all the material such as PCs and cables, for safety and preventing it from being touched by the elderly (Figure 2).

4) **Data transmission and storage:** Vital signs and information on emergency calls are to be sent to a cloud server where a year or newer data are stored. The data older than one year will be archived into a local server. For privacy reasons, no personal data are exchanged between each PC and the server.

5) **User interface (UI):** A specially designed UI is used to monitor each patient’s vital signs and current status. As the UI is also optimized for mobile devices, caregivers can monitor the patients anytime from anywhere without having to sit at a desk with a PC. Additionally, the UI allows keeping a record of what caregivers give to the elderly (Figure 3).

C. **Trial Use at an Actual Care House**

Even at the beginning of the experiment at an actual care house, we could already identify several serious problems and evaluate the effectiveness of the system. In the next section, we will indicate these problems and suggest solutions to overcome them [7].

D. **Overall Architecture of Tele-Monitoring System**

Analog data signal sent from “sensor sheet” will be received by “pre-processor” then converted into digital signal for “Linux computer” to wirelessly transmit the data to our “Cloud server”, where data will be stored. Users can receive...
alerts send out from “web application” and view the vital data of tenants through “personal computers” and “mobile devices”. “IT administrator” would maintain the server and has full access to the database (Figure 4).

III. PROBLEMS

A. Internet Communication Breakdown

Even in our daily lives, Internet connections sometimes get disconnected and are only recovered after resetting the routers or/and computers. This type of disconnection is quite regular, but no one considers the reason of such disconnection. However, if such disconnections occur for a health monitoring system, there is no appropriate person to reset the router or computer. The test site had several disconnection problems; in each case, our staff went to the site to reset the router/computers. This is not a practical method for troubleshooting. A realistic solution may be given by a hardware watchdog system.

a) Hardware for watchdog system: The watchdog system is a well-known concept for embedded computers. The computer periodically sends a specific signal to the watchdog system. If the watchdog system does not receive the signal at the specified timing, it considers that the computer may be inactive, and it forces the computer to restart. We need a complete hardware watchdog system that covers all equipment including sensors, computers, routers, and wireless access points (Figure 4).

All equipment must be on the “power-on reset” mode; hence, when we turn off/on the power supply, the equipment starts from the initial setting before the breakdown.

All equipment must work properly without considering the turn-on timing. Sometimes some sequencing is required; for example, routers should be turned on before the computers. All equipment should work properly by any power-on order. This is necessary in case of adding new equipment to the system; ordered sequence could be a serious obstacle for expandability.

b) Working sequence of the watchdog system: If we use proper equipment, the watchdog system can be as follows. The watchdog system is connected to the router; it is reached from the general administrating computer over the Internet. The general administrating computer is monitoring data uploading, and just after the uploading is successfully done, it sends a signal to the watchdog system of the corresponding dataset. This watchdog system is a simple microcomputer (this microcomputer may also have a watchdog system, to watch a watchdog system) that is running a simple routine code. That is, it sends a turn off/on signal to the power control boxes, by wire or wireless, when it does not receive the specified signal from the general administrating computer for more than five minutes. The power control system is a simple electric circuit installed between the equipment and power outlet, which turns the equipment power supply off and then on when it receives the specified signal.

The entire system works as follows. Typically, all equipment upload their data properly at the specified timing, and the general administrating computer sends the specified signal to the watchdog system as acknowledgement signal. If something happens and the data uploading does not complete successfully, the administration computer does not send the specified signal to the watchdog system; then, the watchdog system turns off and on the power supplies of all equipment. That is, it resets all equipment.

B. Sensor Reliability

Usually, the average failure rate of electric parts is not small enough. There are very low failure rate components for military and space applications, but they are very expensive. For domestic application, we can only use limited reliability components. Meanwhile, we cannot carry out frequent maintenance or on-site replacement of failed components. We plan to carry out the maintenance of the system every three years at minimum. During these three years, we would like to use the system without maintenance even if some part of the system becomes dysfunctional. In this situation, appropriate approaches might be “redundant deployment” and “sensor fusion.”

1) Redundant sensor approach: We install multiple identical sensors redundantly. If the sensors are cheap enough, this approach may be reasonable (Figure 5).

For example, temperature sensors are usually cheap; thus, we can install several temperature sensors in the sheet. Such redundant deployment contributes to increase in the accuracy of measurement as well as fault tolerance. Moreover, we can estimate the body shape in the bed because sensors nearer the body may indicate greater values than sensors far from the body (Figure 6).

2) Sensor fusion approach: The sensor sheet we used at the first trial site is developed by Tanita cooperation, the cost of which is around 400 USD; it can detect heartbeat and breathing rates, standard deviation of body movement, and in/out of bed status.

If we install four weight sensors at the bottom of the bed’s legs for measuring body weight, we can also estimate the breathing rate, heartbeat, and in/out bed status. The accuracy
of these data is inferior to the data of the sensor sheet; nonetheless, we can use these data complementarily when the sensor sheet becomes dysfunctional (Figure 7). Typically, we can use these data in the following manner:

\[ R_i: \text{reliability ratio of the } i\text{-th sensor} \]
\[ D_i: \text{data from the } i\text{-th sensor} \]

where \( R_1 + R_2 + \ldots + R_n = 1, \ 0 \leq R_i < 1 \). \( R_i \) drops to zero when the sensor becomes out of order. These ratios should be determined by long term laboratory testing. Let \( D^* \) be the value that we can use as consolidated data.

\[
D^* = \sum_{i=1}^{n} R_i \cdot D_i
\]

(1)

When the \( i\)-th sensor fails, the ratio will be changed as follows:

\[
D_i \text{ (new)} = 0
\]
\[
D_j \text{ (new)} = \frac{D_j}{1 - D_i}, \quad j = 1 \sim n \text{ except } i
\]

(2)
(3)

The consolidated data can be calculated by the same way. As stated above, we should be able to install several sensors of many categories with multiple communication channels. The data from such sensors should be stored in the database. Hence, the database structure should be flexible and expandable.

C. Huge Database Problem

Our system started smoothly but began to have performance problems after running for a while. The degraded performance occurred when retrieving weeks or months of historical data from the database. Historical data are stored on the disk and their reading is costly if they are stored randomly instead of sequentially. This performance problem is mainly due to the poor schema design, and inappropriate or lacking indexes on a large and growing dataset.

1) Data characteristics: Data are automatically generated from sensors and sent to the server with a time interval of 30 s. The collected sensor data are permanently stored on the database for monitoring and analyzing. Each dataset consists of the readings of heart and breathing rates, body motion, in/out of bed and awake statuses, sensor id, and timestamp; the size is about 109 bytes including the object ID generated by the database. For our experiment, 2,880 sets of data are generated daily from each of the 9 sensor devices located at one facility. Taking into account the characteristics of the data and the potential growth to hundreds or thousands of sensors, the choice of the database became our first consideration for data storing and managing.

2) SQL versus NoSQL databases: Relational databases are designed for consistency, allowing transactions to be easily rolled back if they are not completed properly. Complete consistency is a desirable feature but is not crucial in our system. Envisaging the growing data, partition tolerance is a necessary element; the choice of the database is regarding
where it sacrifices availability or consistency for performance in a distribution system based on the CAP (consistency, availability, partition tolerance) theorem [8]. MongoDB, one of the document-based NoSQL databases, that comes with flexible schema and built-in support for consistency, availability, and horizontal scalability through journaling, replication, and sharding, thereby become our choice. In MongoDB, collections and documents are represented as tables and rows of relational databases, respectively. The document is presented as BSON (binary JSON), which is a binary-encoded serialization of JSON-like documents. JSON (JavaScript object notation) is a lightweight text-based format that is easily interpreted by humans and machines. This format is based on object notation, where object is an unordered set of name/value pairs.

Document for sensor data:

\[
doc = \{ "id" : ObjectId("4f9f71f467b75719ee00002d"), "did" : <device id>, "ts" : ISODate("2012-04-23T01:22:09Z"), "hr" : 72, "rp" : 17, "sd" : 22, "bd" : 1, "wa" : "0" \}
\]

By selecting MongoDB for our read- and write-intensive system, each sensor dataset is stored as a document in a collection and monitored through the Web. The most frequently retrieved data from the database are the recent sensors readings; historical data are less frequently retrieved, unless abnormal readings are found by the monitor, or analysis is needed. We used charts to present sensor readings, with large font numbers representing the current readings of heart and breathing rates and body motion. In addition to the current readings, small font numbers represent the normal range, average, maximum, and minimum on the chart. An image is used to represent the current in/out of bed status (Figure 8).

3) Optimization approach: In order to solve the performance problem mentioned earlier, we redesigned more efficient queries, with the right indexes and schemas in place. Because the number of datasets per day from each sensor is the same, daily data can be grouped by a document and pre-allocated to provide excellent locality and achieve the best disk performance. Obviously, getting the best disk performance also depends on the optimal file system disk configuration. Because our system is running in the cloud, one good choice is the RAID-10 (redundant array of independent disk) disk configuration with ext4 file system as it can provide a fast and contiguous disk allocation. Replication can also be used to overcome some of these performance issues by scaling readings, providing high availability for the system.

Although indexes are crucial for good query performance, they do impose costs such as storage and demand more workload on each write. It is also important to make sure that the working set (data + indexes) fits into the RAM (random access memory) and remove all redundant indexes to lower the chance of page fault. Even with the optimization of indexes, schema, disk performance, and sufficient RAM, our data will eventually no longer fit into the RAM if they continue to grow unbounded. As the data continue to grow, single hardware can no longer handle this growth and horizontal scaling is the next step to solve data size and performance issues [9].

One approach to scale horizontally in MongoDB is called sharding, which is a process for scaling out by partitioning data across more than one machine. Splitting data (collection) up across different machines and keeping the working set in the RAM is the primary reason to shard. This is also a highly effective method for keeping large datasets evenly distributed across multiple machines and improving both read and write performances on large datasets. Combining sharding with replication, MongoDB not only provides a fully consistent, highly available database, but also enables addressing the growth of the unbounded data by just adding more capacity as needed.

D. UI Problems

1) Data analysis: Each person has an individual normal range of heartbeat and breathing rates, and body motion because several factors affect those ranges, such as the activity level, emotion, medication, genetics, gender, body size, air quality, or temperature. To detect abnormal cases and generate valuable messages for caregivers to help the elderly, the system needs to collect sufficient data to assess the average range and daily routine of each individual. If the system does not have enough data, it will use the default formula to detect abnormal ranges. For example, we use 40–100 heartbeats and 10–40 breaths per minute as default normal ranges for every elderly individual.

2) Query data:

a) Frequent information update: Every thirty seconds the system receives new heartbeat and breathing rates, standard deviation of body motion, and in/out of bed status data from each device. Simultaneously, the system detects any abnormal cases for every elderly individual and notifies the caregivers in such cases. When the caregivers start examining an abnormal case, they may go from page to page. The information between those pages must be displayed consistently to show what the problem is.

b) Huge amount of queried data: In order to show a chart of heartbeats over the past three weeks, the system needs to query a huge amount of data. It usually takes time to load the users’ browsers. Depending on the users’ network speed and hardware configuration, the loading time is different. In any
case, the system UI needs to load data within the standard response time (4 s).

3) Simple UI: Caregivers, especially in Japan, tend to keep the old style of keeping the record of their daily care services by hand. That is also a burden of them. To make their job more efficiently, the system UI should be a first step of engaging.

   a) Simplicity: It is the key of the UI design. There are three buttons on the caregiver’s main menu: user profile, logout, and back button. The back button is the main button to go back to the parent page of the current page.

   On each web page, information is just enough to help caregivers performing the job. If there is any critical abnormal case, the system will ask the caregiver to contact the local medical services.

   b) Flexibility: The system UI is multilingual; users can choose their native language in which familiar terminologies are used.

   In every care house, there is a unique regulation; the system UI has to allow managers to create their own environment.

IV. CONCLUSION

In this paper, we described several problems of our health monitoring system for elderly care houses and possible solutions we are going to apply. Our system works perfectly in laboratory-level experiments, but in the case of real deployment, we realized that there are many problems to solve. For example, we have experienced the following situation: A monitoring system of a room was frequently down, the reason being that the system’s AC plug was pulled out from the outlet by dementia elderly. From this experience, we realize that the system should be dementia-proof or at least have a dementia-proof installation.

Another thing we need to mention is that, the project is new and has only been operating for a short while; we are still acquiring enough data to testify the performance and reliability of the system.

This type of research should be rather more pragmatic than other scientific research. Namely, research needs hands-on and experience based approaches. Moreover, we always have to consider the case of large size deployment. In such cases, there are always several unknown and hidden problems, which we never consider during laboratory testing.

The evaluation of research should also be different from the other scientific research.

Standard scientific research is evaluated by how new facts are founded, but elderly care research should be evaluated by how the result contributes to our society.

REFERENCES


