Towards a Peer-to-Peer Communication Model for Mobile Telecare Services

Akio Sashima and Koichi Kurumatani
Human Informatics Research Institute, National Institute of Advanced Industrial Science and Technology (AIST), 2-3-26 Aomi, Koto-ku, Tokyo, 135-0064, Japan

Keywords: Telecare, mHealth, Peer-to-Peer, Mobile Physiological Sensor, Communication Protocol, UDP Hole Punching.

Abstract: In this paper, we describe a peer-to-peer communication model for a mobile telecare service. It is proposed to reduce the service management costs of a conventional mobile telecare service based on a server-client communication model. The peer-to-peer mobile telecare service consists of a mobile physiological sensor, two smartphones, and a connection management server. In the service, when a caregiver, e.g., family member, requires to know current physiological statuses of a cared person, e.g., elderly person, the smartphone directly sends the sensing data of the cared person to the caregiver's smartphone without a central data server. To realize the peer-to-peer communication model in mobile phone's infrastructure, which includes private networks, we propose a communication protocol based on a NAT-traversal technique and a data compression mechanism for preventing packet loss. We have confirmed that the prototype system works well on the current mobile phone's infrastructure that consists of 4G (LTE) and private networks.

1 INTRODUCTION

Recently, mobile health, telemedicine, and telecare services using smartphones have been drawn attention of researchers in scientific and industrial community (Lin, 2012)(Triantafyllidis et al., 2013)(Triantafyllidis et al., 2015). In the mobile health services, physiological and behavior data obtained by wearable smart devices, e.g., watch, phones, glasses, are used for managements of physical and mental conditions (Lu et al., 2012) of the persons who take care of their health.

In mobile telecare services, wearable devices can be used for remotely monitoring the statuses of older persons who live alone in their homes (Sashima et al., 2008)(Vines et al., 2013)(Bellido et al., 2015). The service provides the obtained sensor data for their trustworthy persons who take care about the older person, such as caregivers, family members, doctors, by using their smartphones. Some services can alert them when it detects an emergency situation of the person, such as fall detections (Huang and Newman, 2012)(Sannino et al., 2014).

In this research, we focus on a communication model which is suitable for telecare services using smartphones. So far, in order to implement such telecare services, most of them have been built based on *server-client communication model*; all of sensing data of the older person are sent to a central server and the server manages and provides the sensing data for the community. The model fits for web based services on the Internet very well. For example, sensing data are collected on a web server and the caregivers can see the sensing information, e.g., activity status, as a web content of the web browser that he/she usually uses.

Although a telecare service based on the server-client communication model has been proposed so far, the approach has some drawbacks to start up and maintain a practical service in reality. A drawback is a management cost of the server. For example, using the server-client model, it is hard to prevent overloads in the case of increasing users' traffic accessing the server because all sensing data are necessarily sent to and received from the server. Therefore, to handle the data traffics, it requires that the server has enough information processing power which typical personal computers do not have. In addition, because leakage of the personal data sensed for the telecare service causes privacy issues, managing a lot of personal data in the server rises the management costs.

In order to prevent the issues arising from the server-client model, we propose a *peer-to-peer com-*

Sashima, A. and Kurumatani, K.
Towards a Peer-to-Peer Communication Model for Mobile Telecare Services.
DOI: 10.5220/0005845205420549
In *Proceedings of the 9th International Joint Conference on Biomedical Engineering Systems and Technologies (BIOSTEC 2016) - Volume 5: HEALTHINF*, pages 542-549
ISBN: 978-989-758-170-0

munication model (P2P model) for telecare services. The model assumes that the smartphones of the users directly communicate with each other in peer to peer manner. In other words, the sensing data are directly sent to a smartphone of a caregiver. It is not mediated by a central data server so as to reduce the cost of maintaining the service.

The P2P communication model appears to be appropriate to the telecare service from the maintenance point of view. However, it is unclear that the P2P model works on the current mobile phone's communication infrastructure.

In this paper, we describe a prototype system based on the P2P model which we have developed as a proof of concept and show experimental results measuring communication performance of the system. We focus here on realizing the on-line peer-to-peer communication facility on the current communication infrastructure which includes private networks, and propose a communication protocol based on a NAT-traversal technique. A data compression mechanism for preventing the packet loss of the communication is also proposed. We have confirmed that the prototype system works well on the current mobile phone's infrastructure.

2 COMMUNICATION MODEL OF MOBILE TELECARE SERVICE

In this section, we describe a server-client communication model and a peer-to-peer communication model for implementing a mobile telecare service.

2.1 Server-Client Model

Figure 1 shows an overview of a typical telecare service based on a server client model. In this model, we assume that the service consists of a mobile physiological sensor, two smartphones, and a central data server. The mobile physiological sensor wirelessly communicates with a smartphone. All sensor data from the wireless sensor are sent to a data collection server by a smartphone (sender). The data can be seen by a user with a smartphone (receiver) by accessing the server. It is possible to know the latest status of the user wearing the sensor.

This model can be easily implemented and the server can be stored a large amount of data for the data analysis. However, it has some drawbacks about the management of the server. Because it is hard to prevent overloads at the server in the case of increasing the data traffic flows, it should be a high performance computer that has enough to handle the large traffic loads. In addition, because personal health data on the server should be carefully managed, it also rises the management costs of the server.

2.2 Peer-to-Peer Model

Figure 2 shows an overview of a telecare service based on a peer-to-peer model proposed in this paper. In this model, we assume that the service consists of a mobile physiological sensor, two smartphones, and a connection management server. The mobile physiological sensor wirelessly communicates with a smartphone. All sensor data from the wireless sensor are directly sent to a smartphone (receiver) by a smartphone (sender) using the connection information at the connection management server. The sensed data can be seen by a remote user using her/his smartphone. Hence, it is possible to know the latest status of the user without a central server.

On the current mobile phone's communication infrastructure, however, there are some issues about handling the network address (IP address and port number) for realizing the peer to peer communication. First, the network address of a smartphone is dynamically changed according to its network environments. To enable two smartphones communicate with each other, the network address of the peer is required. Second, the smartphone that connects to a private network with Wi-Fi, such as at home, uses a local IP address. The address is converted to a global address at the network address translator (NAT) router of the network. Similarly, most smartphones of the mobile phone carriers in Japan share and use limited global addresses of the carrier's gateway servers for the communication.

To solve the issues, we introduce 1) a connection management server and 2) a NAT-traversal technique in our peer-to-peer communication model.

2.2.1 Connection Management Server

A connection management server stores the network address of each user's smartphone, and updates them when the smartphone starts to connect to the server. It also tells other smartphones the updated address. By knowing the latest address of the smartphone that they would like to communicate with, the smartphones can communicate each other even if the addresses are dynamically changed.

2.2.2 NAT-Traversal Technique

A NAT traversal technique is used to establish a peer to peer communication channel over a NAT router. It enables a smartphone in a private network communi-

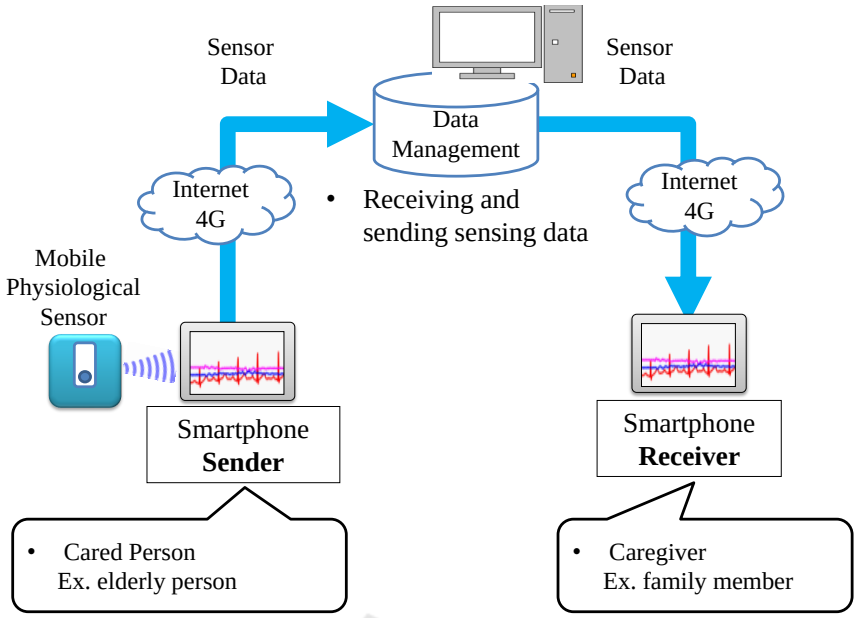

Figure 1: Overview of the client-server communication model.

cating with the smartphone in the outside of the network.

We adopt the UDP hole punching as the NAT traversal technique (Ford et al., 2005)(Rosenberg et al., 2003). The UDP hole punching is a technique establishing a connection for User Datagram Protocol (UDP) packet streams that traverse the NAT. It typically uses the connection management server that enables two smartphones can exchange their network addresses.

In the communication model, a smartphone (A) in a private network sends a UDP packet to the server and waits for the server's response through the NAT router. The server knows the networks address of A by analyzing the received packet and tells the opponent smartphone (B) the address of the A. The opponent smartphone (B) knows the address of the A and sends a UDP packet to the address. The packet is mediated by the NAT router and reaches the smartphone (A) in the private network.

Using the above techniques, we can realize the peer-to-peer communication through the NAT router. Details of the communication protocol are described in Section 3.2.

3 IMPLEMENTATION

In this section, we describe an implementation of a prototype system of the P2P model. It provides a tele–monitoring service of user's statuses, e.g., heart rate, body acceleration, etc. The prototype system consists of a wireless physiological sensor which we have developed (Sashima et al., 2011), two smartphones, and a connection management server. We implement the system on the android devices: Google Nexus 5 and Nexus7.

3.1 Remote Monitoring of Electrocardiographic data

Figure 3 shows a mobile physiological sensor which we have developed. The physiological sensor is a small, wearable, wireless sensor device which includes a flash memory, a lithium ion battery, and 5 kinds of sensors: electrocardiographic sensor (1 channel), 3-axis accelerometer, barometer, thermometer, hygrometer. Its size and weight are as follows: size 6×4×1.5 cm; weight 34.5g. Continuous operating time is about 6–8 hours for sensing electrocardiographic data. It includes a Bluetooth[1] module can be communicate with a smartphone.

In this paper, we have evaluated the validity of the remote monitoring of the Electrocardiographic data using the sensor device.

The device continuously senses electrocardiographic data of the user and wirelessly sends the data to the smartphone. The sensing rate is 200 Hz and the data is represented by 2 byte Integer. The sensor device is attached to user's chest by sticking electrodes with a peel-off sticker.

[1]https://www.bluetooth.com/

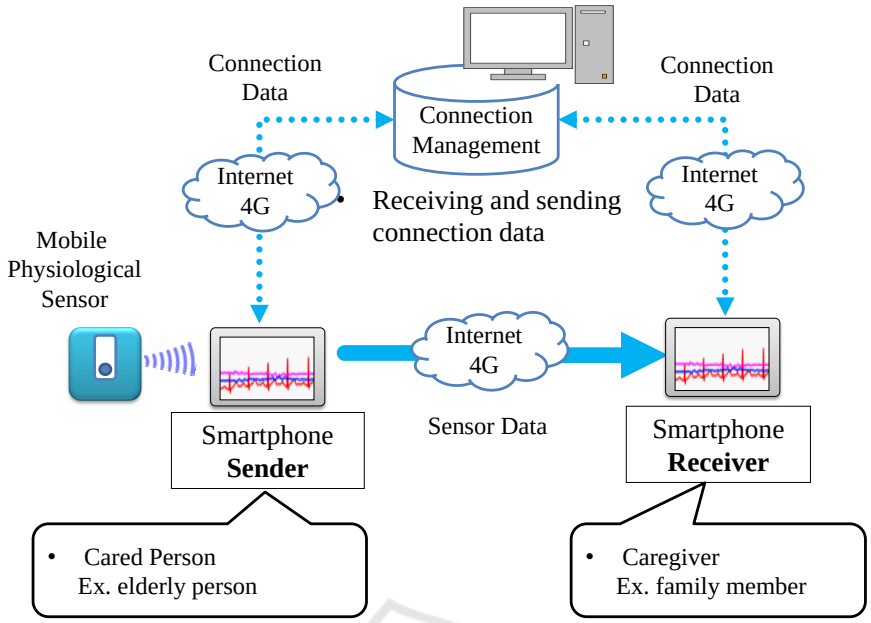

Figure 2: Overview of the peer-to-peer communication model.

Figure 3: Mobile physiological sensor.

3.2 Communication

In current implementation, each smartphone of our system plays a communication role: *sender* or *receiver*. A smartphone playing sender role wirelessly communicates with the sensor device. It sends the sensing data to a smartphone playing receiver role through the 4G network and Wi-Fi. The sender and receiver communicate with the connection server for controlling the peer-to-peer connection. The controlling messages are short texts represented in UDP packets, and exchange the messages with the server for the connection management with NAT-traversal technique.

The server process works on a server machine which has a global IP address on the Internet. It opens a UDP network socket (port) which has a stable port number, 9209, predefined in a service, and waits for the connection messages from users' smartphones. When it receives a message from a user's smartphone, it updates the address of the smartphone, and tells the address to smartphones of the user's companions to enable them accessing the user.

When they know their address each other, it can send and receive the sensing data until one of them disconnects the network. We have implemented two methods for sending the sensing data: 1) a method repeating to send a packet which includes a sensing data, and 2) a method repeating to send a packet which includes a sequence of sensing data with waiting a corresponding ACK packet.

3.2.1 Sending Single Sensing Data without ACK

This method is designed for aiming at fast communications in a stable network environment, such as a local area network (LAN). Figure 4 shows an outline of the protocol using the method. In the methods, each sensing data, such as an electrocardiographic value, received by the sender is wrapped by a UDP packet, and immediately sent to the receiver. Each packet becomes short but the number of the packets are increased. When the receiver receives a packet, it does not send back acknowledge message packets to the sender.

3.2.2 Sending a Sequence of Sensing Data with ACK

This method is designed for aiming at robust communications in noisy network environments, such as

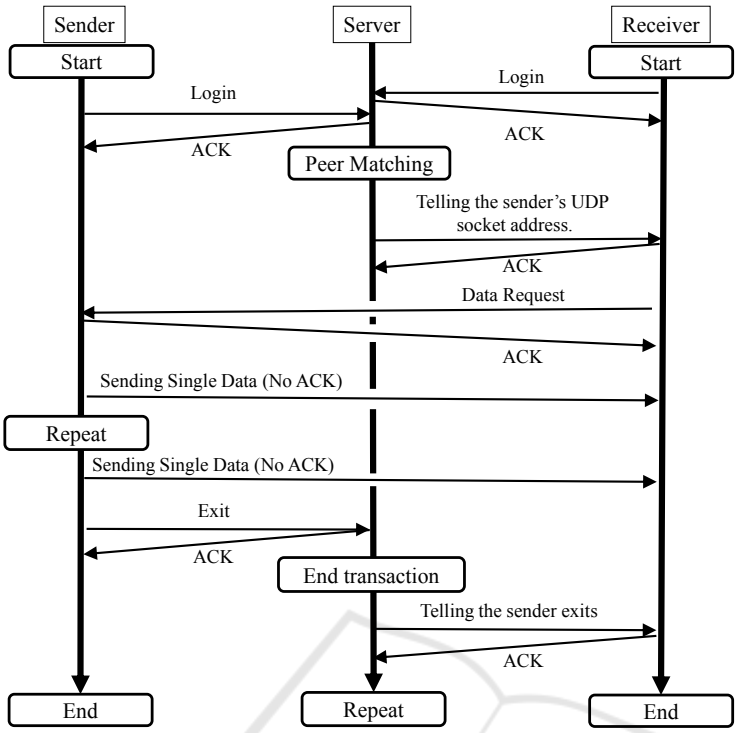

Figure 4: Outline of the protocol of sending raw data without ACK.

wide area network (WAN). Figure 5 shows an outline of the protocol using the method. In the methods, sensing data received by the sender is buffered in a certain time period. Then the buffered data, a sequence of sensing data, is cut and divided into multiple sequences to keep its length within a maximum value predefined. Default value of the maximum length is 120. Each divided sequence is wrapped by a UDP packet, and sent to the receiver. A packet includes a sequence of the data and becomes larger than the packet including single value. Hence, we apply the compression technique to make the packet smaller. "zlib" library[2](Deutsch and Gailly, 1996) is used for the compression. It is a software library based on a compression algorithm called "deflate" (Deutsch, 1996).

When the receiver receives a packet, it sends back an acknowledge (ACK) packet to the sender. The sender sends a next packet when it receives the ACK packet. If it does not receive the ACK packet in a certain time period, it sends the same packet and waits the ACK packets again. If it repeats the sending process five times and still does not receive the ACK packet, it abandons sending the packet and sends a next packet. Although waiting the ACK packet may be a waste of time, it can prevent the packet loss.

[2]http://www.zlib.net/

4 EVALUATIONS

We have experimentally evaluated communication performance of the prototype system. We have specifically investigated the packet loss and delay time between two smartphones, sender and receiver, in different network environments. The network environments are a public 4G (LTE) network provided in Japan, and a Wi-Fi network (LAN), which is a private network behind a NAT router, in our laboratory. We have evaluated the two sending methods: 1) sending single sensing data without ACK, called *raw method* here, and 2) sending a sequence of sensing data with ACK, called *compressed method* here, in four network conditions: A) the sender and receiver connect to the same LAN, B) the sender connects to the LAN and the receiver connects to the 4G network, C) the sender connects to the 4G network and the receiver connects to the LAN, and D) the sender and receiver connect to the same 4G network. We have analyzed the first 100,000 samples received by the receiver for each condition.

4.1 Packet Loss

Experimental results about the packet loss are shown in Table 1. While a few percent of the packet loss oc-

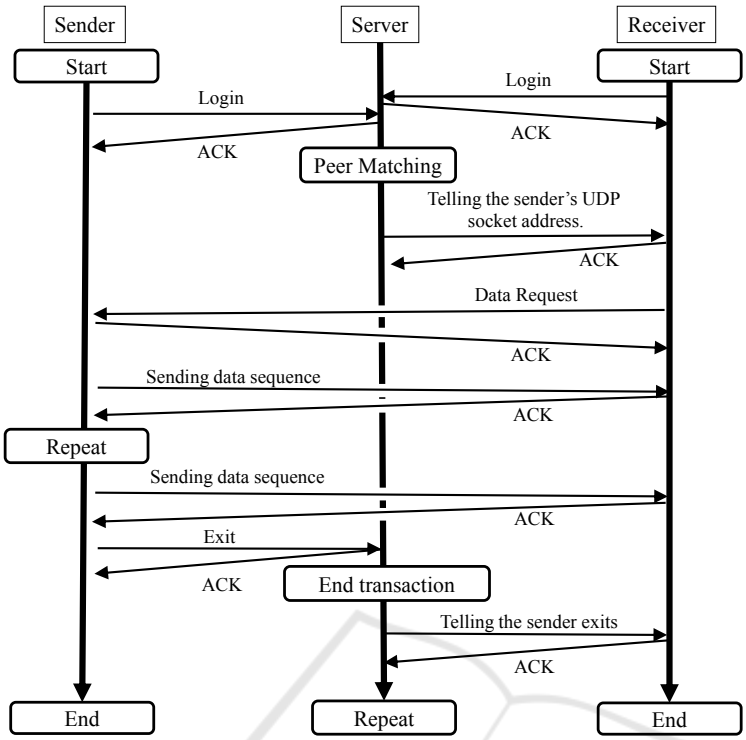

Figure 5: Outline of the protocol of sending data sequence with ACK.

curred by using the raw method, no packets were lost by using the compressed method. We have confirmed that the compressed method prevents the packet loss in NAT-traversal situations.

In the results of the raw method, the worst condition was the condition C in which the receiver in the private network received the packets from the outside sender. It seems that the error is caused by network congestions around the gateway of the private network.

Table 1: Packet loss rate (%).

conditions	compressed	raw
A) LAN	0.0	0.035
B) From LAN to 4G	0.0	0.014
C) From 4G to LAN	0.0	1.626
D) 4G network	0.0	0.020

Figure 6 shows an image of the electrocardiographic data received by the receiver in the condition A. In most conditions, the packet loss in the graph cannot be distinguished by the unaided eye because the packet losses seldom occur.

4.2 Delay Times

We have measured delay times of the communication in different network environments. The results of the raw method are in Figure 7 and the results of the compress method are in Figure 8. Before the experiments the clocks of the sender and receiver are synchronized based on the clock of the connection server. It can be said that they work well in the four conditions: the delay times of the raw method are under 100 msec; the delay times of the compressed method are about 200 msec. Comparing the two methods, the raw method is two times faster than the compressed method. However, delay time of the compressed method is not so bad when we consider their performance of the packet loss. It is considered that the compressed method is useful for most situations for providing the telecare services.

4.3 Changing Packet Size

We have measured the delay times by changing a parameter related to a packet size of the compressed method. The parameter is a value of the maximum length of a sequence in a packet described in Section 3.2.2. In the experiment, the sender and receiver connect to the same 4G network. Figure 9 shows average delay times of the compressed method by changing limit length of the sequence in a packet. The figure shows that changing limited length affects the delay time and the best value is 120. In the experiments, the shortest value, which is 60, has shown the worst

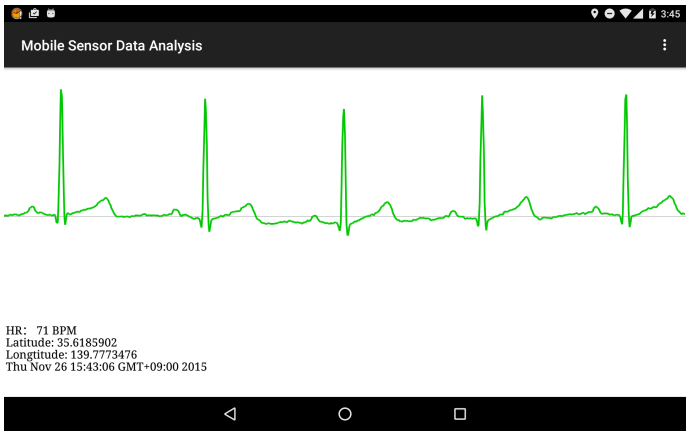

Figure 6: Display image of the received electrocardiographic data.

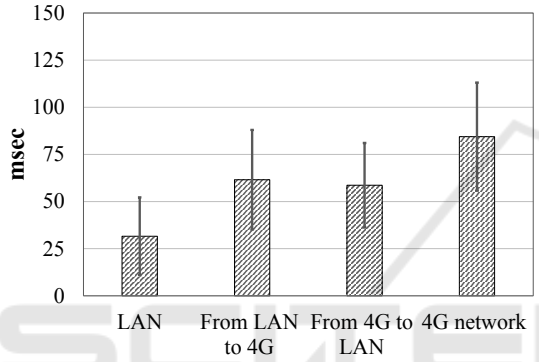

Figure 7: Delay times of the raw method in different network environments.

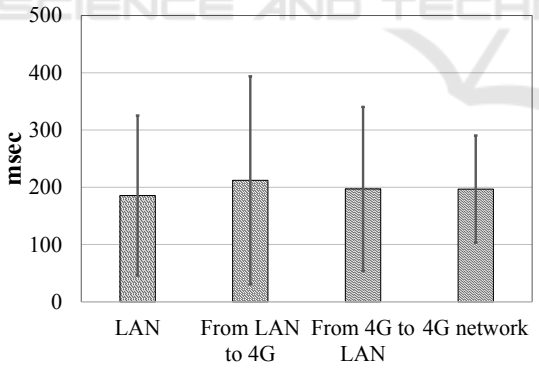

Figure 8: Delay times f the raw method in different network environments.

result.

In the compressed method, the sender waits to send a next packet until it receives an ACK message of a last sent packet. This is a reason of the packet delay. If a packet includes a few data, it takes a lot of time to recover the packet delay because it requires to send many short packets. On the other hand, if a packet with a longer sequence is used, it can recover the packet delay by sending a few packets. Therefore the longer values tend to show the better results in the figure.

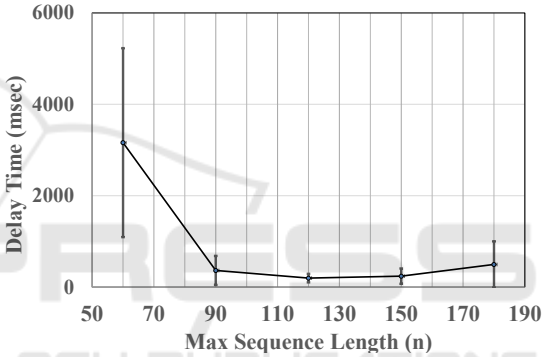

Figure 9: Delay times of the compressed method changing limit length of the sequence in a packet.

5 DISCUSSION AND FUTURE WORK

The experimental results show that the prototype system can establish the communication channel between the user in a private network and the user in the outside of the network. However, it is known that the UDP hole punching technique does not work on some NAT routers, e.g., symmetric NAT, used in enterprise networks. One of our future work is developing a communication method which is adaptable to various network environments, for examples, adopting virtual private network, the combined adoption of server-client model and peer-to-peer model, and so on.

As the prototype system is proposed for a proof of concept of the peer-to-peer communication model, it only uses general techniques for compressing and sending the sensing data. In future work, we are going to develop the method that uses periodical patterns

of the sensing data, e.g., waves of electrocardiogram, for the compression and sending mechanism. We believe that such data-driven methods can realize more efficient communication protocol for the telecare service.

In this paper, we focus on the technical issues of the connection management of the model and do not discuss security issues. Providing a telecare service based on the client-server model, a lot of standard security techniques for managing a Web server, e.g., SSL, can be also used. Providing a telecare service based on the peer-to-peer model, however, there is no standard way to securely manage the service system. Designing a secure peer-to-peer telecare service for practical use will be one of our future work.

6 CONCLUSIONS

We have described a peer-to-peer communication model of a telecare service. The peer-to-peer telecare service model consists of a mobile physiological sensor, two smartphones, and a connection management server. It enables users, such as elderly persons and their caregivers, to share the telecare information, such as electrocardiographic data, without using a central data server of the server-client model. To realize the service in current mobile phone's communication infrastructure, we have proposed a communication protocol based on a NAT-traversal technique and implemented the protocol with a compression mechanism for preventing packet loss. We have confirmed that the prototype system works well on the network environments that include the 4G and private networks.

ACKNOWLEDGEMENTS

This work was supported in part by JSPS KAKENHI Grant Number 26330125.

REFERENCES

Bellido, J. C., De Pietro, G., and Sannino, G. (2015). A prototype of a real-time solution on mobile devices for heart tele-auscultation. In *Proceedings of the 8th ACM International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '15, pages 30:1–30:8, New York, NY, USA. ACM.

Deutsch, L. P. (1996). Deflate compressed data format specification version 1.3. Internet RFC 1951.

Deutsch, P. and Gailly, J.-L. (1996). Zlib compressed data format specification version 3.3. Internet RFC 1950.

Ford, B., Srisuresh, P., and Kegel, D. (2005). Peer-to-peer communication across network address translators. In *Proceedings of the Annual Conference on USENIX Annual Technical Conference*, ATEC '05, pages 13–13, Berkeley, CA, USA. USENIX Association.

Huang, Y. and Newman, K. (2012). Improve quality of care with remote activity and fall detection using ultrasonic sensors. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE*, pages 5854–5857.

Lin, C.-F. (2012). Mobile telemedicine: A survey study. *J. Med. Syst.*, 36(2):511–520.

Lu, H., Frauendorfer, D., Rabbi, M., Mast, M. S., Chittaranjan, G. T., Campbell, A. T., Gatica-Perez, D., and Choudhury, T. (2012). Stresssense: Detecting stress in unconstrained acoustic environments using smartphones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 351–360, New York, NY, USA. ACM.

Rosenberg, J., Weinberger, J., and Mahy-Cisco, R. (2003). STUN-simple traversal of user datagram protocol through network address translators. Internet RFC 3489.

Sannino, G., De Falco, I., and De Pietro, G. (2014). Effective supervised knowledge extraction for an mhealth system for fall detection. In Roa Romero, L. M., editor, *XIII Mediterranean Conference on Medical and Biological Engineering and Computing 2013*, volume 41 of *IFMBE Proceedings*, pages 1378–1381. Springer International Publishing.

Sashima, A., Ikeda, T., Yamamoto, A., Kawamoto, M., Kuga, T., and Kurumatani, K. (2011). Developing mobile physiological sensor that works with indoor positioning system. In *Proceedings of the International Conference on Indoor Positioning and Indoor Navigation*.

Sashima, A., Inoue, Y., Ikeda, T., Yamashita, T., Ohta, M., and Kurumatani, K. (2008). Toward mobile healthcare services by using everyday mobile phones. In *Proceedings of the First International Conference on Health Informatics, HEALTHINF 2008, Funchal, Madeira, Portugal, January 28-31, 2008, Volume 1*, pages 242–245.

Triantafyllidis, A., Koutkias, V., Chouvarda, I., and Maglaveras, N. (2013). A pervasive health system integrating patient monitoring, status logging, and social sharing. *Biomedical and Health Informatics, IEEE Journal of*, 17(1):30–37.

Triantafyllidis, A., Velardo, C., Salvi, D., Shah, S., Koutkias, V., and Tarassenko, L. (2015). A survey of mobile phone sensing, self-reporting and social sharing for pervasive healthcare. *Biomedical and Health Informatics, IEEE Journal of*, PP(99):1–1.

Vines, J., Lindsay, S., Pritchard, G. W., Lie, M., Greathead, D., Olivier, P., and Brittain, K. (2013). Making family care work: Dependence, privacy and remote home monitoring telecare systems. In *Proceedings of the 2013 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, UbiComp '13, pages 607–616, New York, NY, USA. ACM.