Smart Watches: Enrich People's Lives

ZHAO PING
Computer Science Department, University of Auckland
Auckland, New Zealand
pzha291@aucklanduni.ac.nz

ABSTRACT

Watches have been accompanying people's lives for centuries. Smart watches, however, are still relatively new. Compared with smart phones which have significantly changed people's living style in recent years, smart watches still have a long way to go before getting popular. However, in some scenarios such as tough industrial work place and intelligent living environment, smart watches are more likely to be used instead of other devices such as smart phones. In this paper, we review some representative smart watch projects and analyze their approaches. We start by introducing the typical use cases through two generations of smart watches. We then analyze both commonality and differentiation in the methodologies adopted in various approaches. We also discuss the evaluation of such approaches. We finally point out that by providing fast information access and gesture-based interaction with the environment, smart watches may enrich people's lives to a great extent.

ACM Classification Keywords

H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous.

Author Keywords

Smart watch; information access; gesture-based control; gesture recognition; inertial sensor; smart environment

INTRODUCTION

Watches have been invented to accompany people's daily lives for centuries. Traditionally, watches are worn by people as the unique equipment to access date and time information anytime and anywhere. Decades ago, however, with the blooming of cell phones, especially smart phones, people are getting more and more relying on them to handle nearly all sorts of stuff, from simple date and time check to more advanced services such as e-learning and business. That's why for some years, people rarely purchase or wear watches.

This situation hasn't been changed until late of 20 century, when microchip technologies such as microprocessor and Bluetooth experienced significant development. Very quickly, such advanced hardware was adopted into watches area, which inspired the occurrence of smart watches.

IBM's Linux Watch [10] is widely considered to be one the earliest smart watches. At that time, smart watches are mainly designed to use buttons and software menus to interact with users. Their functionalities are mostly focusing on fast information access, such as brief information of a new email, weather forecast, users' PIM (Personal Information Management), and so on. Some other functionalities are focusing on notification through haptic feedback, such as notifying elderly people to take medicine in time.

With the development and application of inertial sensors, smart watches evolved into a new generation, which supports gesture-based interaction with both users and their environment [2]. For instance, in a smart living environment, people need to interact with the smart devices anytime and anywhere. Compared with smart phones which are usually placed on the table at home, smart watches are more likely to be carried everywhere, even in a bathroom, which makes it suitable to be used to control smart appliances and other devices remotely.

The core components in smart watches include a microprocessor, some inertial sensors, and a wireless module. The inertial sensors are critical for the new generation of smart watches to recognize user's gestures and take corresponding actions. From software architecture perspective, smart watches may be designed to work independently, or to work as a client that relies on a remote powerful machine or smart phone for further processing of complicated tasks.

The smart watches of nowadays usually provide users both traditional interaction (such as menus and buttons) and gesture-based interaction, which makes people's lives much easier. Meanwhile, there are still some gaps in this area, such as security protection and standardization, to be considered and filled.

USE CASES

Traditionally, watches are worn by people to access time information anytime and anywhere. This situation hasn't been changed until late of 20 century, when the first generation of smart watches occurred. Since then, watches' information visualization function in a manner of fast access has been significantly extended from sole time information to diverse information that customers may be concerned with. In recent years, powered by inertial sensors, smart watches are able to support gesture-based interaction with both users and their environment in a really smart way.

Fast Information Access

In this information explosion era, people are supposed to access all kinds of information, such as personal emails, social network messages, news, weather forecast, and so on, on a daily or even hourly basis.

Tablet PCs and smart phones have been widely used in terms of information access in personal consumption area. However, neither tablet PC nor smart phone is the best choice to be used to access information quickly in some scenarios. A tablet PC, for instance, is usually placed in a bag and thus not likely to be taken out to check the email when a person is walking on a crowded street. A smart phones, similarly, is usually held in pockets or bags, which increases the risk of missing an important phone call or message even with the vibration notification switched on.

In contrast, smart watches can fill in the gaps very well. User can access multiple types of important information by a simple glance.

Single-Segment Display

Most smart watches have only one segment for display. This is also the case even for those early smart watches which focused their functionalities on fast information access.

IBM's Linux Watch [10] is considered as one of the earliest commercial smart watches. Its prototype is as Figure 1 shows.

Figure 1. IBM's Linux Watch. Reprinted from [10].

The product has two versions, the one with OLED display and the one with LCD display. The only difference between the two versions is just the display. The watch in Figure 1 is of OLED version, which seems to be clearer and more attractive than the LCD version.

Multi-Segment Display

Lyons et al. [8] implement Facet as a wrist worn system consisting of multiple displaying segments. Figure 2 shows the prototype.

Figure 2. The Facet prototype. Reprinted from [8]

Each segment in Facet is meaningful. People can use different segments to store different information, such as Email, and incoming phone calls. Furthermore, powered by Bluetooth and infrared interfaces in some cases, people can easily access Internet services, such as weather forecast, traffic information, stock market, sports news, and so on.

However, what makes Facet really attractive is that people can customize the segments in a direct tangible way. For instance, people can expand one segment display to more ones, or collapse expanded segments to fewer ones. For instance, when someone receives an Email with digest information displayed on one segment, he can expand the display to multiple segments for a further comprehensive reading.

Environment Interaction

With appropriate sensors (such as accelerometers) and wireless modules (such as Bluetooth) integrated, smart watches can be used to interact with both the users and their environments.

Remote Device Control

In some scenarios where it's hard to directly interact with the target due to the limitation in the physical size, people can leverage remote control. In some other scenarios such as a tough industrial environment where smart phones or tablet PCs are not applicable, workers can also rely on smart watches for remote control.

Kim et al. [7] develop Gesture watch, which is a wireless device that controls other devices through hand gestures. The watch first uses sensors to collect raw data of user's gestures, then sent the data over Bluetooth to remote computer or smart phone which will further process the raw data using Markov models. Finally, the recognized hand gestures will be used as commands to control remote devices such as a media player.

One of the recognizable gestures, forward move for one time, is shown in Figure 3.

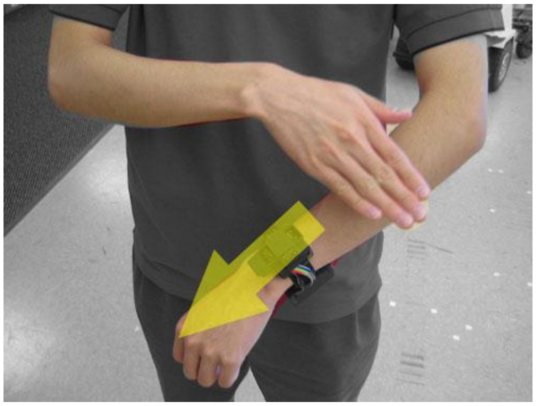

Figure 3. One time forward: move hand from elbow towards fingertips once. Reprinted from [7]

Although tablet PCs and smart phones are popular products for personal consumption, they are not designed to be used in tough industrial environment [2]. Workers usually are required to wear gloves in work place, and their fingers are usually oily, which makes it impossible to operate a smart phone or tablet PC.

Bieber et al. [2] exhibit the feasibilities of smart watches to be leveraged to interact with both the user and his ambient in industrial scenarios. Figure 4 shows a typical usage scenario.

Figure 4. Usage of Smart Watch and Pad in maintenance work. Reprinted from [2].

With the help of a smart watch, the technician is able to control a digital work flow on a PC tablet during his maintenance. The control commands, including start, stop, pause, back, and forward, are interpreted from automatic recognition of the workers activities.

Smart Environment Interaction

Smart environments include various capabilities of a house or building, such as lights, temperature, appliances, entertainment electronics, and so on. Although modern domotic systems (home automation systems) have been considered to simplify the interaction between users and their applications to a great extent, they still have limitations in that they are usually specific devices, and thus not likely to be carried everywhere at home [3].

Researchers have also found that people are not likely to carry their smart phones at home [2]. Instead, they tend to put their smart phones on the table. Then in a smart living environment where intelligent devices need to be controlled through screen based interaction, smart phones are not suitable.

On the contrary, wrist watches are easy to carry, and convenient for control. They are also acceptable and suitable for elderly people who are not familiar with computer system [3].

dWatch is developed by Bonino et al. [3] as a personal wrist watch for smart environment. It has three main functions:

- Provide both stand menus and gesture-based interaction for users to control devices.
- Receive notifications from both the environment and other dWatches.
- Offer personalized functions to users according to their particular privileges. In the cases when multiple dWatches exist, they can share responsibilities with each being granted corresponding privileges to interact with their own devices.

Powered by a domotic gateway, Dog, dWatch is very adaptable and can be easily integrated into various domotic systems.

METHODOLOGIES

In this section, we will analyze the methodologies used by various smart watches from the following aspects respectively.

Hardware

Some smart watches are based on existing commercial watches, such as MetaWatch used in [2], WIMM One used by Facet [8], and eZ430-Chronos used by dWatch [3]. One of the advantages of leveraging existing watches is that there's no need to design or customize hardware components.

However, current commercial watches sometimes cannot meet the requirements of particular research projects. That's why many research works in smart watches tend to choose appropriate hardware components and assemble them together to make them work efficiently.

Here we discuss some common components used in both commercial watches and research-based watches.

Microprocessor

The microprocessor is the core component in smart watches. Depending on what tasks the watches are supposed to do, various microprocessors are adopted.

MSP430, from Texas Instruments, is one of the earliest microprocessors applied in smart watches [2]. The watches in [2] and [6] are all based on it.

Some high-end microprocessors, such as ARM 7 in [10] and ARM 9 in [9], can provide considerable computation abilities, which are suitable for watches that process complicated tasks, such as gesture recognition.

Sensors

Gestures recognition in smart watches is usually implemented through inertial Sensors. There are three common inertial sensors: accelerometers, gyroscopes, and magnetometer.

- Accelerometers are cheap solutions which provide relative position information by combing the sensors values twice. Thus they are normally used to recognize swipe, shake, and tilting gestures. The drawback is that they cannot provide absolute position information without support from outside [9] [4].
- Gyroscopes are high-end solutions which provide relative orientation information by calculating the angular rate. Combined with other sensors such as magnetometers, they are possible to provide absolute orientation information. They are usually used to detect rotation of the forearm [4].
- Magnetometer is used to recognize tilting gestures and pointing gestures [9].

Beside inertial sensors, there are other sensors, such as infrared proximity sensors in [7] and force sensors in [9], used in different watches.

Wireless Module

Wireless module is a regular configuration for most smart watches. The most prevalent wireless protocol is Bluetooth which is used by [6] [10] [7] [9]. Meanwhile, there are some other wireless protocols, such as SimpliciTI in [3] which is used and modified to allow dWatch to communicate with a domotic gateway.

Software Architecture

Depending on how computational tasks are distributed, the software architecture of smart watches can be categorized into three types: all-in-one, client-server, and gateway-integration.

All-in-one

This type of smart watches usually have powerful processing abilities that can handle intensive computational tasks, such as gestures recognition. Normally they independently complete all the tasks, such as sensing data collection, gestures recognition, commands issuing, interaction with users, and so on.

Many current smart watches belong to this category. E.g., MetaWatch [2], Facet [8], IBM's Linux Watch [10], and the smart watch developed in [9] all independently recognize user’s gestures and other input, interact with users, and do some management stuff.

Client-Server

This type of smart watches usually has limited processing abilities. Instead of handling all the tasks by themselves, they usually send the raw sensing data to remote server for further processing, such as gesture recognition. In most cases, they are only responsible for information display and simple interaction with users.

Personal Server (PerServ) proposed by Hutterer et al. [6] is a typical example of client-server architecture, where the watch is only used for display purpose, including receiving, decoding and displaying raw information. All the computational intensive tasks are left to PerServ. The architecture is as Figure 5 shows.

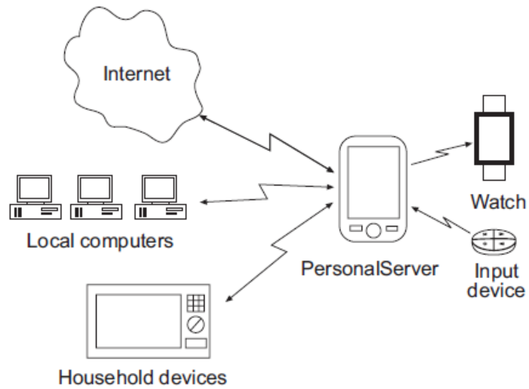

Figure 5. Overview of Personal Server Design. Reprinted from [6].

On the other side, PerServ runs on a standard HP iPAQ which is powerful enough to do complicated computation.

Another example is Gesture Watch [7], which sends the collected sensing data to remote computer or smart phone for further processing, including gestures recognition.

Gateway-Integration

In some cases smart watches cannot directly interact with remote devices. Instead, they have to communicate with some gateways first, and then the remote devices.

dWatch [3] is integrated into a smart environment through Dog, a domotic gateway that enables integration of different home automation solutions. To complete the integration, Bonino et al. [3] develop a new device driver, which is installed in Dog gateway and used to communicate with dWatch over wireless protocol. Then dWatch can interact with remote devices through Dog.

The whole work flow is that, dWatch first collects the accelerometer data, which is then sent to gesture recognition module. Once the gesture is recognized, the corresponding command is sent to Dog gateway and then the remote devices.

Interaction

The old generation of smart watches provides users menu and button interfaces, whereas the relatively advanced smart watches enable users to interact with them base on gestures. Besides, there's another type of interaction, tangible gesture interaction, which is still in concept verification stage at this moment.

Menu & Button Interface

With IBM's Linux Watch [10], Users can interact with the watch through both touch screen (software buttons at four corners) and roller wheel (top right), as Figure 1 shows. The top left button includes start menus. The bottom left button enables users to return back. The top right button serves as an alarm setting. The bottom right button can lead users to the phone number list. The roller wheel, on the other hand, enables users to switch between applications.

dWatch [3], based on eZ430-Chronos, has 5 buttons for users interaction, as Figure 6 shows.

Figure 6. The eZ430-Chronos watch and USB dongle. This watch is what dWatch is based on. Reprinted from [3].

The star button (*) is used to navigate through menus. The hash button (#) is for configuring a selected menu. Bonino et al. [3] also develop some software menus, such as Time, Alarm, Temp, Gesture, and so on, to be run on dWatch, Time is a basic function. Alarm allows people to send an alarm notification to other dWatches. Temp shows users the temperature of environment, and allows user to control temperature by pressing the up-arrow or down-arrow button. Gesture menu allows users to enable and disable the gesture-based control of devices.

Gesture-based Interaction

The new generation of smart watches is usually integrated with inertial sensors, and thus has the abilities of recognize user's gestures, which makes gesture-based interaction possible. E.g., MetaWatch [2], Facet [8], and dWatch [3] all support gesture-based interaction.

Facet [8] even implements multi-touch interaction, which is very rare among devices with small-size screens, such as those of smart watches. Although Facet segment do not individually support multi-touch, two or three fingers touching separate screen at the same time forms multiple segments touch. During that, Facet first gains the data of relative touch position, and then the pose of the whole bracelet. Finally the number of touches and position of each touch is sensed.

On the other hand, these smart watches still inherit and keep traditional interaction method, such as menus and buttons, from the old generation. For instance, dWatch [3] allows users to choose whether enable gesture-based control, and provides users buttons to access menus, as Figure 6 shows.

Tangible Gesture Interaction

HOVEN et al. [5] proposed an intersection research of gesture interaction and tangible interaction. Traditional gesture research only considers empty-handed gestures, while hand-held objects can be used as representations and controls, and thus be able to enhance or change people's gesture. This is such a new area that very few researchers have touched it. Therefore, lots of researches need to be done in this area.

EVALUATION

The fast information access being able to bring great conveniences for people has been widely used and evaluated consequently. In contrast, gesture-based interaction is still a new area, which may need more evaluations to be made.

So far, some smart watch researches have revealed their evaluations, some of which are quite quantified and thus more likely to be reliable. E.g., the usage of prototype watch developed in [2] shows that hands-free interaction through wrist watch improves the quality of work, and makes the maintenance easier. The experiments of Gesture watch [7] show that 95.5% accuracy is gained, and there are no obvious differences between standing and walking, indoor and outdoor environment.

SUMMARY

Smart watches have been experiencing significant development in recent years. A large amount of applications have been developed to extend the scenarios where smart watches can be used, which has also inspired the related researches.

The old generation of smart watches are focusing on fast information access, whereas the new generation of smart watches are focusing on gesture-based interaction with their environments.

Microprocessors, sensors, and wireless modules are core components for smart watches. To adapt what types of components, however, depends on what the tasks are, and how the tasks are distributed.

In HCI aspects, the new generation of smart watches usually combine traditional menus and buttons interfaces with advanced gesture-based interaction.

Many smart watches are still in either prototype development or verification stage, thus evaluations are not sufficient to some extent. However, some cases have shown their convincing experiment results.

FUTURE WORK

Even though smart watches have experienced significant development in recent years, there are still some gaps waiting to be filled in the future.

Standardization

Unlike general software engineering area, almost all smart watches that support gesture recognition have developed their own set of gestures, which on the one hand results in a huge waste of money and time. On the other hand, this hinders the technique communication between research communities, and thus hinders the progress of the development of smart watches as a whole.

Fortunately, not all the research projects overlook such a huge gap. Ferscha et al. [4] propose and develop a general framework to detect orientation based gestures. The core of the framework is to standardize the gestures, and form a general gestures library, which includes three types of gestures: hand gestures, gestures of permanent hand-held objects, and gestures of occasional operation of hand-held objects.

Security Protection

People usually wear watches on the wrist with the displaying panel outside oriented, which may lead to the leakage of their personal information. Although this is a serious issue, it seems to be overlooked by most smart watches. Fortunately, it may be solved in watches that have multi-segment displays. Lyons et al. [8] have suggested that applications should consider the position of display in order to protect privacy.

To protect sensitive information from being stolen, Al-Muhtadi et al. [1] proposes a context-based security mechanism that provides identification and authentication of users through automated reasoning. This may be adopted by those smart watches that usually access user's sensitive information.

Battery Life

Battery life is one of the main concerns that most smart watches are faced with [10] [3]. For instance, WIMM ONE, a popular commercial smart watch, can only standby up to 36 hours [3].

Normally higher computational needs mean more consumption of battery, whereas a lower computational requirement such as the watch in PerServ architecture [6] may significantly reduce the power consumption. However, these watches must rely on other devices for further process, which may not be a good idea for those high-end smart watches that seek to handle all the tasks by themselves.

Therefore, there's still a long way for smart watches to go to achieve a long battery life without losing high computational abilities.

REFERENCES

1. Al-Muhtadi, J., Ranganathan, A., Campbell, R., and Mickunas, M.D. Cerberus: A Context-Aware Security Scheme for Smart Spaces. *Pervasive Computing and Communications*, IEEE CS Press (2003), 489-496.

   Electronic Link:
   http://ieeexplore.ieee.org.ezproxy.auckland.ac.nz/stamp/stamp.jsp?tp=&arnumber=1192774

2. Bieber, G., Kirste, T., and Urban, B. Ambient Interaction by Smart Watches. In *Proceedings of the 5th International Conference on PErvasive Technologies Related to Assistive Environments*, 39, ACM Press (2012).

   Electronic Link:
   http://delivery.acm.org.ezproxy.auckland.ac.nz/10.1145/2420000/2413147/a39-bieber.pdf?ip=130.216.158.78&acc=ACTIVE%20SERVICE&CFID=195545163&CFTOKEN=15196893&__acm__=1364010175_5f1532722ade8acd67a65a578c5bbc43

3. Bonino, D., Corno, F., and Russis, L. D. dwatch: A personal wrist watch for smart environments. *Procedia Computer Science*, *10*, Elsevier Press (2012), 300-307.

   Electronic Link:
   http://ac.els-cdn.com/S1877050912003973/1-s2.0-S1877050912003973-main.pdf?_tid=c4dd4392-9f7c-11e2-91ef-00000aab0f01&acdnat=1365337080_a8c12e49e6a53305c779979f590c8819

4. Ferscha, A., Resmerita, S., Holzmann, C., and Reichor, M. Orientation sensing for gesture-based interaction with smart artifacts. *Computer Communications*, *28*(13), Elsevier Press (2005), 1552–1563.

   Electronic Link:
   http://ac.els-cdn.com.ezproxy.auckland.ac.nz/S0140366405001301/1-s2.0-S0140366405001301-main.pdf?_tid=4d549454-a163-11e2-afe8-00000aab0f01&acdnat=1365546044_24b7159edef4ebe598638df1e7000be

5. HOVEN, E.V.D., and MAZALEK, A. Grasping gestures: Gesturing with physical artifacts. *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, *25*, Cambridge University Press (2011), 255–271.

   Electronic Link: http://www.elisevandenhoven.com/publications/hoven-aiedam11.pdf

6. Hutterer, P., Smith, M.T., Thomas, B.H., Piekarski, W., and Ankcorn, J. Lightweight user interfaces for watch based displays. In *Proceedings of the Sixth Australasian conference on User interface*, *40*, Australian Computer Society, 2005, 89–98.

   Electronic Link: http://delivery.acm.org.ezproxy.auckland.ac.nz/10.1145/1090000/1082255/p89-hutterer.pdf?ip=130.216.158.78&acc=PUBLIC&CFID=306840186&CFTOKEN=62537505&__acm__=1364813614_02b3b89cd171aa95f82eec7eb892db37

7. Kim, J., He, J., Lyons, K., and Starner, T. The Gesture Watch: A Wireless Contact-free Gesture based Wrist Interface. Wearable Computers, In *Proceedings of the 2007 11th IEEE International Symposium on Wearable Computers*, IEEE (2007), 15-22.

   Electronic Link: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4373770

8. Lyons, K., Nguyen, D.H., Ashbrook, D., and White, S. Facet: A Multi-Segment Wrist Worn System. In *Proceedings of the 25th annual ACM symposium on User interface software and technology*, ACM Press (2012), 123-130.

   Electronic Link: http://delivery.acm.org.ezproxy.auckland.ac.nz/10.1145/2390000/2380134/p123-lyons.pdf?ip=130.216.158.78&acc=ACTIVE%20SERVICE&CFID=195538716&CFTOKEN=89562440&__acm__=1364009971_e617dc9550e1fd745eab3dc5f224c69f

9. Morganti, E., Angelini, L., Adami, A., Lalanne, D., Lorenzelli, L., and Mugellini, E. A Smart Watch with Embedded Sensors to Recognize Objects, Grasps and Forearm Gestures. *Procedia Engineering*, *41*, Elsevier Press (2012), 1169-1175.

   Electronic Link: http://ac.els-cdn.com/S1877705812026975/1-s2.0-S1877705812026975-main.pdf?_tid=b0a579ca-a0e5-11e2-ad66-00000aab0f6b&acdnat=1365492094_377a24b4f513b9103f0c822f9ef41d58

10. Narayanaswami, C., Kamijoh, N., Raghunath, M., Inoue, T., Cipolla, T., Sanford, J., Schlig, E., Venkiteswaran, S., Guniguntala, D., Kulkarni, V., and Yamazaki, K. IBM's Linux watch, the challenge of miniaturization. *Computer*, *35* (1), 2002, 33-41.

    Electronic Link: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=976917