Literature Review:
An investigation into the usefulness of the Smart Watch Interface for university students and the types of data they would require

Kyle Mills Johnson

30 May, 2014

Supervisors: Professor Hannah Thinyane, Mrs. Ingrid Siebörger
Department of Computer Science, Rhodes University

Abstract

The aim of this literature review is to investigate and present the smart watch as a potentially useful device for information access used by university students. In this review the smart watch interface, which is part of wearable computing, is looked at as an alternative data access platform to the already existing and non-wearable smart phone. Current application usage and development on the smart phone and smart watch interfaces, as well as the information which university students would want from such devices are discussed. Hardware and software advancements on the smart watch which could be driving access to user information are also covered.

Contents

1 Introduction

With the introduction of the smart watch interface to the family of wearable computing devices, the question has been posed as to whether smart watches can provide their users with useful applications. Smart watches are now seen to be socially acceptable in the modern digital world, and can possibly be used as an alternative interface for information access (Bieber, Haescher, & Vahl, 2013). Smart watches which were previously available on the market offered a limited number of applications at the discretion of manufacturers, with no intention of taking advantage of any open source software solutions (Smith, 2013). The standardization of Application Programming Interfaces (API's) and the increasing availability of libraries through Software Development Kits (SDK's) for these devices, means that new applications can be created relatively quickly by developers and are able to meet the needs of users more dynamically than in the past (Sachse, 2010).

Applications for smart watches are being developed on open platforms such as *Android* which has been seen with Sony's *Open Smartwatch* project (Sony, 2014a) and Pebble's open development project (Pebble, 2014). As seen by the findings of Narayanaswami and Raghunath (2000) as well as Hutterer, Smith, Thomas, Piekarski, and Ankcorn (2005) there have already been investigations into the need to use such a device over already established platforms and what data the users of these devices would want. Further studies regarding the usability of smart watches have also been conducted by Pascoe and Thomson (2007). This literature review will investigate the smart watch as a possible alternative interface to the smart phone for the access of data desired by university students.

2 Wearable Computing

Modern day wearable computing began in the 1950's with work done by Edward Thorp who developed a small device which could be concealed in a shoe. This device analyzed the motion of a roulette wheel via micro-switch pulses and predicted which portion of the wheel the ball would land on (Thorp, 1998). Thorp's device was seen as tackling a single problem domain, where all resources on the wearable device were aimed at solving only one particular problem. As a result, many new wearable computing devices were developed in the 1970's which tackled single purpose problem domains only.

Single domain solutions were created for areas in Computer Science. One such device which solved problems in computer vision was the *WearComp* device developed by Mann (1997a). This device categorized how scenes responded to different lighting situations and shot pictures accordingly (Mann, 1997a). A head mounted camera was also developed for the domain of computer vision, which viewed two dimensional objects as if they were in three dimensions by using two miniature

CRT monitors in front of the user's eyes (Sutherland, 1968). Situational or contextual awareness was addressed in devices such as the *Olivetti Active Badge*, which provided the locations of users who wore them by receiving an infra-red signal from transceivers installed in specially wired buildings (Greaves, 2000; Want, Hopper, Falcao, & Gibbons, 1992). This approach to solving problems in a single domain was found to be too restrictive for a general purpose computing system, and so efforts were made to create devices which could solve problems in multiple domains (Starner, 2001).

According to Rhodes (1997), generalized wearable computing devices (devices which solve problems in multiple domains) should have the following characteristics:

a) Portable while remaining operational, thereby allowing a user to move and still operate the device.
b) Allow for hands free use and allow for non-obstructive access.
c) Integrate sensors such as wireless communications, cameras, GPS, microphones and accelerometers as input devices to provide information about the close environment.
d) Communicate information to the user in a proactive way, thus conveying information to the user even when not being actively used. An example would be alerting the user when a new email has arrived.
e) Always being on and continuously receiving information about the surrounding environment.

The points which Rhodes puts forward are evident to varying degrees in wearable computers taking a more generalised approach to problem solving.

The convergence of multiple problem domains being dealt with by a single programmable device emerged in the early 1980's with a back-pack sized multimedia and text based system which was able to record video and positional information (Mann, 1997b). This inspired further research into augmented reality devices, in which three dimensional virtual objects are added into environments in real time (Azuma, 1997). Augmented reality approaches were used in military, entertainment and medical applications in conjunction with existing technologies to solve problems in more than one problem domain (Azuma, 1997; Mann, 1997b).

The multi-programmable approach to augmented reality in wearable computing resulted in the United States Military using an experimental wearable computing system with their troops, which incorporated a wireless communication link and a helmet-mounted display to aid in battlefield tasks (Zieniewicz, Johnson, Wong, & Flatt, 2002). This popular technique of generalizing the wearable computer to solve a broad range of problems through augmented reality spurred Carnegie Mellon University to develop *VuMan* in the early 1990's, which was a wearable computer that was far more general purpose than its predecessors (Bass, Kasabach, Siewiorek, & Smailagic, 1997). These early systems did not find any commercial success as they were cumbersome and not easily accessible, resulting in a lack of social acceptance (Starner, 2001).

Miniaturization of hardware components, the availability of low cost sensors and the existence of widespread Internet access over the last few years, have allowed wearable computing devices to become more commonplace, readily wearable and socially acceptable (Swan, 2012). This vigorous expansion of miniature computing devices has led to the concept of *The Internet of Things (IOT)* which can be described as a collection of interconnected and interactive devices which are able to communicate useful real-time information between one another (Swan, 2012). These devices are able to incorporate multiple sensors and vastly improved resources to provide users with convenient access to information relevant to a broad range of applications (Narayanaswami & Raghunath, 2000).

Some notable examples driving the idea of the *IOT* are: *Google Glass*, *Fitbit* and the *Nike FuelBand*. *Google Glass* (Google, 2014) has an optical head-mounted display providing users with a natural language and gesture controlled interface to allow for augmented reality. The *Fitbit* (Fitbit, 2014) and *Nike FuelBand* (Nike, 2014) extend the idea of interconnectedness, where the systems consist of a centralized base-station (smart phone) and a wristband interface which uses an integrated accelerometer to measure movement in a three dimensional space (Montgomery-Downs, Insana, & Bond, 2012). The smart watch also falls into the category of the *IOT*, in which it acts as a peripheral device to a connected smart phone (Bieber et al., 2013). The smart watch will be the primary focus of this literature review, and how it can be used in this interconnected way to solve some of these generalized problems which wearable computing devices have been developed for.

3 The Smart Watch

Using Rhodes (1997) definition of wearable device requirements, it can be seen that the smart watch meets these characteristics. The smart watch is: a) portable while still being accessible from the user's arm, b) allows for hands free operation, c) integrates sensors as input devices, d) communicates information to the user despite not being actively instructed to do so and e) continuously receives information on the surrounding environment. Starner (2013) has stated the smart watch should also allow for the integration of computer processing in everyday life, allowing users to retrieve information no matter where or when they wish to access it. Smailagic (2002) has also said the smart watch needs to be aware of user context to be able to respond appropriately in particular situations. It is important that the hardware fits into a user's lifestyle without being an obstruction, and allow for human-computer interaction to occur with minimal cognitive overhead (Starner, 2013). Marks (2013) found that smart watches provided these attributes to their users by being an accessible and convenient method for accessing information; thus smart watches fit the requirements for wearable computing devices. In this section I will discuss the advantages and disadvantages of the smart watch interface, together with a description of one of the smart watches currently available on the market, the *Pebble*.

3.1 Advantages

Smart watches allow for the convenience of always available information access on the wrist. These devices have the advantage of always being with the user due to their wrist-watch form factor, which allows for information access at the flick of the wrist, and also makes it less likely to be misplaced (Narayanaswami & Raghunath, 2000). Unlike the smart phone interface, smart watches allow a user to have both hands free thus enabling the user to be unhindered while accessing information (Pascoe & Thomson, 2007). Bieber et al. (2013) found that it aided users in tracking physical movements through the use of the watch's accelerometer, where the force exerted on the watch could be used in gesture controlled applications.

The wristwatch form factor is already socially accepted, where users are familiar with the commonly available wristwatch and feel comfortable wearing them (Kim, He, Lyons, & Starner, 2007). Like the common wristwatch, the interface provides a minimalistic approach to information access, and the user can choose what information they want without having to process unnecessary data (Angelini, Caon, Carrino, & Bergeron, 2013). Although the screen sizes on smart watches are small, their monochromatic nature such as seen on the *Pebble* provides users with a strong textual contrast to allow for strain-free viewing of information. Ye, Malu, Oh, and Findlater (2014) found that visually impaired users preferred to view information on these monochromatic displays rather than on the common smart phone interface.

Recent smart watch designs have resulted in more efficient power consumption (Bieber et al., 2013). This improved power efficiency has aided in the convergence of services into a single device which are based on sensory data and user context (Starner, 2001). These services differ from those incorporated into the smart phone interface as they are suited to particular user contexts to solve problems (Starner, 2001).

3.2 Disadvantages

Despite the advantages which smart watches present, there are still some noticeable drawbacks and limitations which need to be addressed. Hutterer et al. (2005) and Narayanaswami, Kamijoh, Raghunath, Inoue, and Cipolla (2002) have found that power consumption was a significant limitation in the smart watch architecture; this was due primarily to the Graphical User Interfaces (GUI's) and Operating Systems (OS's) used. Smart watches which are not architecturally designed to conserve power, still have to be charged regularly in order to be used (Narayanaswami et al., 2002).

Since the findings of Hutterer et al. (2005), there has been a steady increase in the complexity of GUI's and OS's used on smart watches, which has led to the improvement of battery technology. However, the improvements in power conservation and extended battery life continue to lag behind computational requirements (Bieber et al., 2013). In some designs, this has resulted in a more centralised approach to data processing. Using such a design allows most data to be processed by a single base station and not on the smart watch itself (Bieber et al., 2013). This processed data can then be communicated back and forth to the smart watch as a peripheral device to the base station (Bieber et al., 2013). The use of low powered monochromatic displays in newer, more minimal derivations of the smart watch also appear to mitigate the negative effects of increasing power consumption (Bieber et al., 2013). However GUI interaction remains limited to a small touch screen and a few external buttons on smart watches with monochromatic displays. Smart watches such as the *Samsung Galaxy Gear 2* incorporate power intensive features such a touch screen display, camera and heart rate monitor (Samsung, 2014). The use of these features continue the pattern of increasing battery capacity to scale with higher computational requirements (Ping, 2013).

The miniaturised screen on the smart watch can be seen as both an advantage and disadvantage. As stated earlier, Ye et al. (2014) found that the monochromatic screen and side button approach of the Pebble was described as being favourable to visually impaired individuals. However, on smart watches which allow for a touch screen interface, users have commented on the limitations imposed by their fingers obscuring the interface (Perrault & Lecolinet, 2014).

3.3 The Pebble Smart Watch

The *Pebble* is powered by an *ARM* Cortex-M3 processor and runs its own *Pebble* OS, which is a customized version of *FreeRTOS*. It differs from its competitors, such as the *Samsung Galaxy Gear 2* and *Sony SmartWatch 2*, in that it is not a stand-alone device and requires pairing with a mobile phone within its Bluetooth range (Bieber, Kirste, & Urban, 2012; Samsung, 2014; Sony, 2014b). Data processing is not done on the *Pebble* smart watch itself but rather sent to the paired mobile phone (Bieber et al., 2013).

This paired architecture between the watch and base station allows for the *Pebble* to be charged only once every few days (Bieber et al., 2013). Furthermore, the *Pebble* utilizes a monochromatic display which has an ambient light sensor that regulates the screen brightness depending on lighting conditions (Bieber et al., 2013). Despite the incorporation of a three-axis accelerometer and a magnetometer, which provides data input based on the surrounding environment, the Pebble still manages to maintain power efficiency (Chyla, 2013). The use of an OS which depends on few hardware resources, and an exterior design incorporating four buttons for human-computer interaction, means user interaction with the *Pebble* is kept simple and intuitive (Narayanaswami & Raghunath, 2000).

The *Pebble* presents a multi-programmable platform on which applications written for *Android* can be executed. This general purpose wearable computing solution incorporates many of the advantages of wearable computing such as convenience and always accessible data retrieval, while providing limited drawbacks when compared to other smart watches such as power consumption and complicated user interfaces.

4 Comparisons to other current mobile interfaces

Human-computer interaction techniques with wearable and mobile devices was found to be important when users selected to use one device over another (Starner, 2013). This included the physical ease with which the device could be used and interacted with, and the ease of access to desired information (Starner, 2013). According to Rukzio, Leichtenstern, Callaghan, and Holleis (2006), the context of the user is also important when assessing device preference, as some situations may make certain interfaces more accessible than others. Devices such as *Google Glass*, smart watches and other mobile platforms have their own unique attributes which separate them from one another in how they access information, and can differentiate them in terms of user preference (Smailagic, 2002).

The smart watch takes the form of a standard wristwatch and thus allows for an unobtrusive and always viewable source of information without the necessity of reaching for any other external device (Marks, 2013). Some smart watches are currently viewed as complimentary devices to smart phones, where both devices synchronize their operations to complete tasks. Some smart watches fit this paradigm more closely than others, while some smart watches maintain an independent data processing model. For example, the *Samsung Galaxy Gear 2* is designed to operate as more of a standalone device, which is capable of synchronizing periodically to a smart phone, as opposed to the *Pebble* smart watch which communicates constantly to a paired smart phone (Bieber et al., 2013). These synchronization options have implications when assessing processing power on mobile devices.

In terms of processing power on mobile devices, the smart phone has been found to exhibit a greater ability to deal with computationally intensive tasks than its other wearable counterparts (Marcial, 2010). A more powerful processor is able to be implemented on the smart phone due to the smart phone having a more powerful battery than most wearable devices can currently support (Bieber et al., 2013). Users have been found to derive benefits from familiar screen layouts when interacting with devices (Marcial, 2010). The smart phone provides a familiar interface to users through the use of icons and menus (Marcial, 2010). However, the smart phone is not specifically designed for problems which wearable devices aim to solve (Starner, 2001).

Wearable devices such as *Google Glass* are primarily concerned with augmented reality specific tasks, and are used in a hands-free way to retrieve data. The *Google Glass* interface allows for an always available head-mounted display above the eye, and can be interacted with via voice commands (Google, 2014). Applications for *Glass* have been used largely in the medical field, where data can be retrieved and synchronized to an external device. Levine (2014) described the ease of use for applications in the field of Obstetrics such as the streaming of vital signs during patient sedation and for streaming live ultrasound images. There was also a usage found for *Glass* in pediatric surgery by Muensterer and Lacher (2014), where relevant data required by surgeons was easy to access, and surgical procedures were able to be recorded while a simulated surgical procedure was performed. *Glass* has the advantage of convenience in allowing information retrieval without any arm movement, but has been seen to lack the necessary processing power to deal with any computationally intensive tasks and multitasking functionality (Muensterer & Lacher, 2014).

Smart phones, smart watches and *Google Glass* have the ability to synchronize their actions with a base-station for further processing or storage purposes (Bieber et al., 2013). These interfaces are designed to be used for specific tasks where the convenience of information access can be maximized depending on user context (Smailagic, 2002).

5 Application Development

The adoption of smart watches by users was addressed by Seppala and Broens (2013), where it was discovered that the experience of retrieving information was the key principle driving information access through any particular device. User experience was also affected by how users could control their personal flow of information. The way in which users retrieve and control their information is through the applications which have been developed for smart watches. Application development for the smart watch has currently been pursued by *Android* and *Apple iOS* for the open source and proprietary markets respectively. These applications are initially installed on a smart phone where they are able to interact with a connected smart watch via Bluetooth.

5.1 Current mobile applications used by university students

According to Bowen and Pistilli (2012) mobile applications can be divided into the categories of:

- Device neutral, which are delivered to the device from a mobile browser; and
- Native applications, designed for a specific platform such as *Android* or *Apple iOS*.

The choice between which category of application is used, has been shown to be dependent upon the activity being performed and the nature of the information needed (Bowen & Pistilli, 2012). Application usage amongst university students has been found to be dependent upon students' lifestyles (Sandars, Homer, Pell, & Croker, 2008). The development of native applications appears to model the popularity of these lifestyle information requirements (Sandars et al., 2008).

Undergraduate university students have been found to use native applications more readily than device neutral applications accessed through a mobile web browser (Bowen & Pistilli, 2012). A study into the applications which were popular amongst undergraduate medical school students was conducted by Sandars et al. (2008), in which it was discovered that users were primarily involved with social networking, email and instant messaging based applications. The same users were found to use device neutral applications in circumstances where information was of a less personal nature. These device neutral applications are designed for widespread access among multiple users, such as online content sharing websites and wikis (Kreutzer, 2009).

Connectivity has been found to be an important aspect when determining what applications are being used by university students (Sandars et al., 2008). Students have been found to use web based services primarily to download native applications for their devices and for communication services (Sandars et al., 2008). These native applications are subsequently used via the mobile web to perform social networking, shopping, news, education and location based services (Bowen & Pistilli, 2012). Native applications most commonly used without the use of the mobile web were found to be gaming, music and media based services (Bowen & Pistilli, 2012).

Software used on most smart watches is run on the paired smart phone, acting as a native application, with the exception of the *Samsung Galaxy Gear* variants (Chyla, 2013). The smart watch is then used as an external device for gaining access and generating any input data that can be processed further on the phone. Application usage on the smart watch has yet to be studied as in depth as its smart phone counterpart. However, sensor-based data on user context and information accessibility appear to be driving application usage on the smart watch (Chen, Grossman, Wigdor, & Fitzmaurice, 2014). Information can be easily accessed on the wrist while the user is engaged in an activity where it would be inconvenient to access information on a smart phone (Bieber et al., 2013). The wristwatch based design has also made time based applications such as the use of stopwatches, alarms and date information popular amongst users (Narayanaswami & Raghunath, 2000).

From studies done by Chen et al. (2014) it is evident that users want similar services from a smart watch as those they get from a smart phone. The method of information access is the primary aspect which has changed to facilitate convenience. Since users want similar applications from different devices, application development for both devices must adapt to meet a standardized output (Chen et al., 2014).

5.2 Application development comparison with the smart phone platform

Application development for increasingly powerful mobile phone hardware is currently supported by OS models such as *Windows Phone 8*, *Apple iOS* and *Android* (Meier, 2012). *Windows Phone 8* and *Apple iOS* are closed source and can prioritize applications depending on whether they are native or third-party, restrict inter-process communications and ultimately limit the control of data between user and system space in the OS (Meier, 2012). *Android* takes a different approach to partially eliminate such restrictions by employing an open source OS model and using a *Linux* kernel (Meier, 2012). The open source nature and common application structure of *Android* has also made it popular for smart phone application development (Joseph, 2013). The *Android* OS will be investigated for its development potential on both the smart watch and smart phone platforms.

Android development on smart phones has become increasingly popular. According to the current smart phone market share, *Android* represents approximately 37% of the market (Liu, Li, Guo, Shen, & Chen, 2013). The study conducted by Bowen and Pistilli (2012) (n=1566) also found that 43% of the students who partook in their user study used *Android* devices, followed by *Apple iOS* (40%) and Other (17%). *Android* applications are developed via a *Java* based SDK and an *Android* Development Tools plug-in for the *Eclipse*[1] Integrated Development Environment (IDE) which can be installed from the developer website (Android, 2014). These development tools include the necessary API's and libraries for the creation of *Android* smart phone applications. Applications developed using the *Android* SDK can be run on the integrated emulator or flashed to a connected *Android* device for testing and debugging (Sheusi, 2013).

The popularity of *Android* native smart phone applications is evident from Perez (2014) where it was found that *Android* applications were downloaded 45% more than their *Apple* counterparts in the first quarter of 2014. From the user study conducted by Bowen and Pistilli (2012), popular native applications used for social networking such as *Facebook*, *Twitter* and *Instagram* already have multiple variations for users to chose from. According to Bowen and Pistilli (2012) there are two approaches to developing native applications for a smart phone: standard native development and the use of online mobile development frameworks such as *PhoneGap*[2] and *Appcelerator*[3]. These online development tools allow for the creation of applications using a toolset of web development languages such as *CSS3*, *JavaScript* and *HTML5* (Bowen & Pistilli, 2012). Native device functions are still able to be accessed through a *JavaScript* API between the OS and the container running the application on the web (Charland & Leroux, 2011).

Device neutral applications or cloud-based services are less dependent upon device resources

---

[1] http://www.eclipse.org/

[2] http://phonegap.com/

[3] http://www.appcelerator.com/

and run on a client-server architecture, allowing for access to information via a mobile Internet connection (Dinh, Lee, Niyato, & Wang, 2013). The development of device neutral applications is not specific to *Android*, the only prerequisite is the installation of a *HTML5*, *JavaScript* and *CSS3* capable browser on the smart phone (Dinh et al., 2013).

The application development for the smart watch follows a similar approach to that of the process used to create native smart phone applications. The main differences being the device used to install and debug the application on, and the use of a different SDK in *Eclipse*. The *Android Wear* SDK is a set of specific API's and libraries for wearable devices which use the *Android* platform (Wear, 2014). This SDK focuses on integrating unique wearable computing functionality, such as the use of wearable sensors, into the development of *Android* applications (Wear, 2014). Development for the *Pebble* smart watch requires *Pebble's* own SDK (*PebbleKit*), which is available for installation on *Linux* and *MacOS X*, as no *Android* OS runs natively on the *Pebble* smart watch itself (Bieber et al., 2012). *Pebble* also provide a web based development framework through *CloudPebble* for use on *Windows* machines (CloudPebble, 2014).

There are distinctions between smart watch and smart phone development (Chen et al., 2014). The *Android* OS on the smart phone is able to utilize more of the available on-system hardware resources than the OS on the smart watch. This is a limitation for application development on smart watches such as the *Samsung Galaxy Gear 2* and *Sony SmartWatch 2* which have native *Tizen* and *Android* OS kernels respectively (Samsung, 2014; Tizen, 2014). Other smart watches implement a proprietary kernel, which is used to retrieve and send data to and from the smart phone (Bieber et al., 2013). This allows for a simpler approach to application development, where all development is performed on the smart phone, just as native smart phone applications are, and not on the smart watch (Bieber et al., 2013). Allowing the smart watch to become a peripheral device to the smart phone means the native kernel on the smart watch does not require access to a large number of on-device resources (Bieber et al., 2013).

5.3 Possible future smart watch advancements

The emergence of devices which solve new problems, and allow for data access in new user contexts will expand the family of wearable computing devices (Abowd, Dey, Orr, & Brotherton, 1998). According to Chen et al. (2014), this expansion involves smart devices, including smart watches, being interconnected to each other. Starner (2001) has previously stated that devices which satisfy user requirements most adequately will survive in the wearable computing market. According to Marks (2013), the smart watch interface has been predicted to replace the smart phone in accomplishing simple tasks such as the viewing of text and accessing sensory data. The growing popularity of the smart watch amongst users has already led to innovative solutions in problem areas involving gesture control and movement based analysis (Bieber et al., 2012; Morganti et al., 2012). With the continuing improvements of hardware and software being developed for the smart watch interface, it is expected to grow in popularity (Schlegelmilch, 2014).

5.3.1 Exploiting new hardware

The lack of rapid change in smart watch hardware growth is due partly to the interaction between the smart watch and smart phone, where the operation of the smart watch is largely dependent on the more powerful smart phone (Patterson, 2013). This observed slow change is also partly due to Bell's Law which states that a new platform emerges approximately once every decade, and is based on a new interface and form of usage which establishes a new industry (Bell, 2014). A new platform, such as that of the smart watch, will have an initially slow growth in performance characteristics in comparison to the smart phone (Bell, 2014). It is expected that the need for further changes in smart watch hardware will become more commonplace as the platform becomes more popular (Patterson, 2013). Smart watches which are designed to perform more on-device computations in the future will likely have multiple processor cores incorporated into their designs (Horowitz, 2014). Such is the case with the *Samsung Galaxy Gear 2* (Samsung, 2014) and *Neptune Pine* (NeptunePine, 2014) which already incorporate multi-core CPU architectures. Besides the obvious hardware components of the smart watch interface, individual components such as the battery, display and radio transmitters / receivers have the potential to be improved upon in order to add to the overall functionality of the smart watch (Horowitz, 2014).

The ability to supply power to the smart watch device without negatively affecting user experience is still an ongoing challenge (Hodges, 2013). Battery technology on smart watches continues to lag behind the improvements of hardware resources which follow Moore's law (Hodges, 2013). The majority of current mobile computing devices use rechargeable Lithium-ion (Li-ion) batteries which struggle to scale with ever increasing computational demands (Hodges, 2013). Currently the effects of this are mitigated by designing hardware as efficiently as possible (Weyland, 2013). Emerging technologies such as carbon-fiber wearable super-capacitors aim to store a combination of energy from environmental sources such as body heat, sunlight, body movement and ambient radio frequencies (Jost, Stenger, Perez, & McDonough, 2013). Until such technologies come into the mainstream market, components which consume the most power like the display and radio transmitters will have to be designed with maximum power efficiency in mind (Horowitz, 2014; Weyland, 2013).

Alternatives are being found for current display technologies on smart watches which divert from the common liquid crystal (LCD) and monochromatic displays currently used (Mims, 2013). *Qualcomm* have developed a *Mirasol* display which operates by refracting light and requires no power unless being updated (Mims, 2013; Qualcomm, 2014). The monochromatic memory LCD display technology used on the *Pebble* is the same as that used by the *Agent* smart watch (Agent, 2014). This screen technology is being pushed as the future for minimal display technologies. Memory LCD's are being used primarily for peripheral screens which compliment a main LCD display being used on a device such as a smart phone (Mims, 2013). With the design of efficient display technologies and other components, new features are able to be incorporated into the smart watch, such as additional radio connectivity (Horowitz, 2014; Weyland, 2013).

Near Field Communication (NFC) is already being used for digital wallet applications on smart phones via the use of specialized chips which are based on existing Radio Frequency Identification (RFID) standards (McHugh & Yarmey, 2014). According to Goss (2013) NFC applications on smart watches could be one of their most desirable features, creating a potential "killer app" which could take advantage of the convenience of an always available wrist based interface and user context. NFC capability has yet to become a common feature on most current smart watches with only the *Sony Smartwatch 2* currently capable (Sony, 2014a), as well as a lack of widespread use on smart phones. The convenience of the smart watch paired with NFC highlights the need for some degree of interconnectedness between smart watches and the *IOT* (Goss, 2013).

With the growing nature of the *IOT*, smart watches are set to join smart environments which consist of multiple interconnected devices performing micro-interactions via radio technologies such as RFID and Bluetooth (De Russis, 2014; Weyland, 2013). An example of such an environment would be that of household automation systems, which are able to be controlled from a device such as a smart phone to interact with physical objects in an environment. Automation systems being developed by *iControl*[4] plan on incorporating the paired connection architecture of the *Pebble* and smart phone (De Russis, 2014). In such an architecture, the smart phone acts as a gateway to access the web and consequently is able to control actuators in a physical domain via a web service (De Russis, 2014).

Smart watches which present the right balance between hardware performance, extended battery life, interaction with their environments, and the use of applications which improve user experience will ultimately lead the smart watch interface into the future (Patterson, 2013). Hardware advancements provide a solid foundation for the OS's being run on them, including any Input/Output (I/O) software which is developed. I/O has been found to be a challenge on small devices, both as a hardware and software issue, and often requires unique implementation techniques (Funk, Sahami, Henze, & Schmidt, 2014).

5.3.2 Proposed new input and output techniques

When harnessing the power of recent hardware advances, possibilities begin to arise as to what information can be accessed through that hardware (Narayanaswami, Raghunath, & Kamijoh, 2001). I/O software being developed for smart watches attempt to take advantage of, and compliment the hardware available (Bieber et al., 2013). Since the smart watch has limitations associated with its small screen size such as text input and data output techniques, there have been

---

[4] http://www.icontrol.com/

attempts to alleviate these issues (Funk et al., 2014). Research has been conducted in building onto existing hardware and software available for the smart watch to improve I/O techniques. (Funk et al., 2014).

For small screen devices such as the smart watch, Dunlop, Komninos, and Durga (2014) found an easier way to input text in email, messaging and social networking applications. The new method uses an optimized on-screen alphabetic layout, similar to those currently used on smart phones, and can be used on smart watches with a touch screen interface (Dunlop et al., 2014). Funk et al. (2014) developed a wristband based text input system which has replaced the wrist-strap of the watch. This solution uses multiple sensors in the watch strap which are mapped to visual elements on the surface of the strap, such as function keys and letters for text entry Funk et al. (2014). Wrist-strap input aims to remove the problem of screen occlusion caused by text input via the user's finger on touch screen displays Funk et al. (2014). Amma and Schultz (2013) have developed a hand gesture method for text input, which uses the hand of the user as a stylus and a character recognition algorithm. User interaction with smart watches involving flexible input and rapid text entry are vital components of the smart watch interface (Wobbrock, 2006). The way the user interacts with the smart watch in terms of both input and output has much to do with the overall user experience (Wobbrock, 2006). Other means to allow for the input of information have been researched such as gesture based interaction and voice commands.

Gesture controlled input for smart watches has been studied by Bieber et al. (2012), in which it was found that situational awareness and the use of sensors could be used as input to the smart watch device. A study performed by Xiao, Laput, and Harrison (2014) studied user input using wrist gestures where an application was developed that used pan and tilt movements to navigate different menus and select various icons. Voice commands are now being used as the primary input method in some smart watches such as the *Martian*(Martian, 2014), which allows for communication with *Google Now*[5] and *Apple Siri*[6] on the connected smart phone. The use of voice as an input has limitations in terms of clarity for the smart watch to recognize in noisy situations, however attempts at implementing noise cancellation into the device's microphone have proven to be successful (Shanklin, 2013). Both voice and gesture control software have been implemented as a standard feature on the *Kreyos Meteor* (Kreyos, 2014) smart watch. The *Kreyos Meteor* has the functionality to answer calls at the wave of a hand, and can be programmed to perform any number of actions when interacted with by a specific wrist gesture (Garber, 2013).

Data output techniques on smart watch displays have to be developed with physical size constraints in mind (Chen et al., 2014). Attempting to display too much information on devices with small screens can raise the problem of text overflow onto multiple screens or information clipping (Sanchez & Goolsbee, 2010). *Spritz* text reading software is now available, which allows the user to maintain the viewing of a fixed position on the screen, while text enters the view area from right to left for the user to read (Spritz, 2014). This method of reading text has the advantage

---

[5] http://www.google.com/landing/now/

[6] http://www.apple.com/ios/siri/

of only allowing a few characters to be displayed at any time on the screen, thus saving screen space (Spritz, 2014). However, the *Spritz* technique used for reading has the disadvantage of a possible loss of context while reading, this is due to there being no past reference kept of the text being read (Paul, 2014). The *Samsung Galaxy Gear 2* has the software already installed, however the SDK is freely available from the *Spritz* website for developers to implement on other devices (Petrovan, 2014). Output on the smart watch is not only limited to visual data representations, but also haptic and aural via the use of vibration motors and internal speakers respectively.

6 Summary

This literature review has discussed the smart watch interface within the domain of wearable computing with specific focus on the *Pebble*. The *Pebble* has been compared to with other smart watches currently available such as the *Samsung Galaxy Gear 2*, *Sony SmartWatch 2*, *Kreyos Meteor*, *Neptune Pine*, *Martian* and the *Agent*. The limited disadvantages on the *Pebble* were also discussed, such as the minimal display, efficient power management and simple user interaction techniques used. The smart watch has been analyzed as an alternative interface to the conventional smart phone, and the differences between these interfaces have been addressed. Convenience and always accessible data access have been found to be the driving factors for the popularity of the smart watch interface amongst users.

Previous studies found that university students prefer to use native applications over web based services on smart phones. Issues relating to how university students interacted with these devices for information access, and issues affecting information needs on such devices have been highlighted. This literature review has reviewed relevant issues relating to smart watch usage, such as how information accessibility and user context contribute to its growing popularity. It has also been found that users want similar information from smart watches as they do from smart phones.

References

Abowd, D., Dey, A. K., Orr, R., & Brotherton, J. (1998). Context-awareness in wearable and ubiquitous computing. *Virtual Reality*, *3*(3), 200–211.

Agent. (2014). *Agent smartwatches.* Retrieved from `http://agentwatches.com/` (Accessed on 28/05/2014)

Amma, C., & Schultz, T. (2013, December). Airwriting: Bringing Text Entry to Wearable Computers. *XRDS*, *20*(2), 50–55.

Android. (2014). *Android developers.* Retrieved from `http://developer.android.com/develop/index.html` (Accessed on 28/05/2014)

Angelini, L., Caon, M., Carrino, S., & Bergeron. (2013). Designing a desirable smart bracelet for older adults. In *Proceedings of the 2013 ACM Conference on Pervasive and Ubiquitous Computing Adjunct Publication* (pp. 425–434). New York, NY, USA: ACM.

Azuma, R. T. (1997). A survey of augmented reality. *Presence*, *6*(4), 355–385.

Bass, L., Kasabach, R., Siewiorek, D., & Smailagic. (1997). The design of a wearable computer. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems* (pp. 139–146). New York, NY, USA: ACM.

Bell, G. (2014, January). *Moores law evolved the pc industry; bell's law disrupted it with players, phones, and tablets: New platforms, tools, and sevices* (Tech. Rep.). Microsoft Research, Silicon Valley Laboratory 835 Market Street. Suite 700 San Francisco, CA 94103: Microsoft Corporation.

Bieber, G., Haescher, M., & Vahl, M. (2013). Sensor Requirements for Activity Recognition on Smart Watches. In *Proceedings of the 6th International Conference on Pervasive Technologies Related to Assistive Environments* (pp. 67:1–67:6). New York, NY, USA: ACM.

Bieber, G., Kirste, T., & Urban, B. (2012). Ambient Interaction by Smart Watches. In *Proceedings of the 5th International Conference on Pervasive Technologies Related to Assistive Environments* (pp. 39:1–39:6). New York, NY, USA: ACM.

Bowen, K., & Pistilli, M. D. (2012). *Student Preferences for Mobile App Usage.* (Forthcoming)

Charland, A., & Leroux, B. (2011, May). Mobile Application Development: Web vs. Native. *Commun. ACM*, *54*(5), 49–53.

Chen, X. A., Grossman, T., Wigdor, D. J., & Fitzmaurice, G. (2014). Duet: Exploring Joint Interactions on a Smart Phone and a Smart Watch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 159–168). New York, NY, USA: ACM.

Chyla, P. (2013). *Remote display for smartphone* (Unpublished master's thesis). California State University Northridge.

CloudPebble. (2014). *Cloud pebble.* Retrieved from `https://cloudpebble.net/` (Accessed on 28/05/2014)

De Russis, L. (2014). *Interacting with Smart Environments: Users, Interfaces, and Devices* (Unpublished doctoral dissertation). Politecnico di Torino.

Dinh, H. T., Lee, C., Niyato, D., & Wang, P. (2013). A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless Communications and Mobile Computing*, *13*(18), 1587–1611.

Dunlop, M. D., Komninos, A., & Durga, N. (2014). Towards High Quality Text Entry on Smartwatches. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (pp. 2365–2370). New York, NY, USA: ACM.

Fitbit. (2014). *Fitbit.* Retrieved from `http://www.fitbit.com/` (Accessed on 27/05/2014)

Funk, M., Sahami, A., Henze, N., & Schmidt, A. (2014). Using a Touch-sensitive Wristband for Text Entry on Smart Watches. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems* (pp. 2305–2310). New York, NY, USA: ACM.

Garber, L. (2013). Gestural Technology: Moving Interfaces in a New Direction [Technology News]. *Computer*, *46*(10), 22–25.

Google. (2014). *Glass.* Retrieved from `http://www.google.co.za/glass/start/` (Accessed on 27/05/2014)

Goss, P. (2013, June). *Why NFC is the killer app for the smartwatch.* Retrieved from `http://www.techradar.com/news/world-of-tech/your-laziness-will-force-nfc-onto-apple-watch-1143721` (Accessed on 21/05/2014)

Greaves, D. (2000). *Olivetti research active badge.* Retrieved from `http://koo.corpus.cam.ac.uk/projects/badges/` (Accessed on 27/05/2014)

Hodges, S. (2013). Batteries Not Included: Powering the Ubiquitous Computing Dream. *Microsoft Research Cambridge*, *46*(4), 90-93.

Horowitz, M. (2014). 1.1 Computing's energy problem (and what we can do about it). In *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2014 IEEE International* (pp. 10–14).

Hutterer, P., Smith, M. T., Thomas, B. H., Piekarski, W., & Ankcorn, J. (2005). Lightweight User Interfaces for Watch Based Displays. In *Proceedings of the Sixth Australasian Conference on User Interface - Volume 40* (pp. 89–98). Darlinghurst, Australia, Australia: Australian Computer Society, Inc.

Joseph, J. (2013). Mobile OS–Comparative Study. *Journal of Engineering Computers & Applied Sciences*, *2*(10), 10–19.

Jost, K., Stenger, D., Perez, C. R., & McDonough. (2013). Knitted and screen printed carbon-fiber supercapacitors for applications in wearable electronics. *Energy & Environmental Science*, *6*(9), 2698–2705.

Kim, J., He, J., Lyons, K., & Starner, T. (2007). The gesture watch: A wireless contact-free gesture based wrist interface. In *Wearable Computers, 2007 11th IEEE International Symposium on* (pp. 15–22).

Kreutzer, T. (2009). Generation mobile: online and digital media usage on mobile phones among low-income urban youth in South Africa. *University of Cape Town, SA.*

Kreyos. (2014). *Kreyos meteor.* Retrieved from `https://www.kreyos.com/` (Accessed on 28/05/2014)

Liu, Y., Li, F., Guo, L., Shen, B., & Chen, S. (2013). A comparative study of android and iOS for accessing internet streaming services. In *Passive and Active Measurement* (pp. 104–114).

Mann, S. (1997a, Oct). An historical account of the 'WearComp' and 'WearCam' inventions developed for applications in 'personal imaging'. In *Wearable Computers, 1997. Digest of Papers., First International Symposium on Wearable Computing* (p. 66-73).

Mann, S. (1997b). Smart clothing: The wearable computer and wearcam. *Personal Technologies*, *1*(1), 21-27.

Marcial, L. H. (2010). *A comparison of screen size and interaction technique: Examining execution times on the smartphone, tablet and traditional desktop computer* (Unpublished master's thesis). University of North Carolina at Chapel Hill.

Marks, P. (2013). Samsung launch kickstarts the smartwatch boom. *New Scientist*, *219*(2934), 22.

Martian. (2014). *Martian smartwatch.* Retrieved from `http://www.martianwatches.com/` (Accessed on 28/05/2014)

McHugh, S., & Yarmey, K. (2014). Near field communication: Recent developments and library implications. *Synthesis Lectures on Emerging Trends in Librarianship*, *1*(1), 13.

Meier, R. (2012). *Professional Android 4 application development.* John Wiley & Sons.

Mims, C. (2013, July). *What I learned from researching almost every single smart watch that has been rumored or announced.* Retrieved from `http://qz.com/102646/takeaways-from-every-single-smart-watch/` (Accessed on 21/05/2014)

Montgomery-Downs, H., Insana, S., & Bond, J. (2012). Movement toward a novel activity monitoring device. *Sleep and Breathing*, *16*(3), 913-917.

Morganti, E., Angelini, L., Adami, A., Lalanne, D., Lorenzelli, L., & Mugellini, E. (2012). A Smart Watch with Embedded Sensors to Recognize Objects, Grasps and Forearm Gestures . *Procedia Engineering*, *41*(0), 1169 - 1175. (International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012))

Muensterer, O. J., & Lacher, M. (2014). Google Glass in pediatric surgery: An exploratory study . *International Journal of Surgery*, *12*(4), 281 - 289.

Narayanaswami, C., Kamijoh, N., Raghunath, M., Inoue, T., & Cipolla. (2002, Jan). IBM's Linux watch, the challenge of miniaturization. *Computer*, *35*(1), 33-41.

Narayanaswami, C., & Raghunath, M. (2000). Application Design for a Smart Watch with a High Resolution Display. *2012 16th International Symposium on Wearable Computers*, *0*, 7.

Narayanaswami, C., Raghunath, M., & Kamijoh. (2001, January). *What would you do with a hundred mips on your wrist?* (Tech. Rep.). Thomas J. Watson Research Center P.O. Box 218 Yorktown Heights, NY 10598: IBM Research Division.

NeptunePine. (2014). *Neptune pine smartwatch.* Retrieved from `www.neptunepine.com` (Accessed on 28/05/2014)

Nike. (2014). *Nike+ Fuelband SE.* Retrieved from `http://www.nike.com/us/en_us/c/nikeplus-fuelband` (Accessed on 27/05/2014)

Pascoe, J., & Thomson, K. (2007). On the Use of Mobile Tools in Everyday Life. In *Proceedings of the 19th Australasian Conference on Computer-Human Interaction: Entertaining User Interfaces* (pp. 39–47). New York, NY, USA: ACM.

Patterson, S. M. (2013, October). *How smartwatch designers should be designing smartwatches.* Retrieved from `http://www.networkworld.com/community/blog/how-smartwatch-designers-should-design-smartwatches` (Accessed on 21/05/2014)

Paul, A. M. (2014, March). *The problem with that speed reading app everyone is talking about.* Retrieved from `http://theweek.com/article/index/258243/the-problem-with-that-speed-reading-app-everyones-talking-about` (Accessed on 28/05/2014)

Pebble. (2014). *Develop for pebble.* Retrieved from `https://developer.getpebble.com/` (Accessed on 27/05/2014)

Perez, S. (2014, April). *Google play still tops ios app store downloads, and now narrowing revenue gap, too.* Retrieved from `http://techcrunch.com/2014/04/15/google-play-still-tops-ios-app-store-downloads-and-now-narrowing-revenue-gap-too/` (Accessed on 29/05/2014)

Perrault, S., & Lecolinet, E. (2014). Using Low-Power Sensors to Enhance Interaction on Wristwatches and Bracelets. In G. Memmi & U. Blanke (Eds.), *Mobile Computing, Applications, and Services* (Vol. 130, p. 261-264). Springer International Publishing.

Petrovan, B. (2014, March). *Coming on the gear 2, spritz could make reading emails on a smartwatch a breeze.* Retrieved from `http://www.androidauthority.com/spritz-speed-read-samsung-galaxy-s5-gear-2-356274/` (Accessed on 21/05/2014)

Ping, Z. (2013). *Smart Watches: Enrich Peoples Lives* (Unpublished doctoral dissertation). University of Auckland.

Qualcomm. (2014). *Mirasol.* Retrieved from `http://www.qualcomm.com/mirasol` (Accessed on 28/05/2014)

Rhodes, B. (1997). The wearable remembrance agent: A system for augmented memory. *Personal Technologies*, *1*(4), 218-224.

Rukzio, E., Leichtenstern, K., Callaghan, V., & Holleis. (2006). An Experimental Comparison of

Physical Mobile Interaction Techniques: Touching, Pointing and Scanning. In P. Dourish & A. Friday (Eds.), *UbiComp 2006: Ubiquitous Computing* (Vol. 4206, p. 87-104). Springer Berlin Heidelberg.

Sachse, J. (2010). *The standardization of Widget-APIs as an approach for overcoming device fragmentation* (Unpublished master's thesis). HTW-Berlin (University of Applied Sciences).

Samsung. (2014). *Galaxy Gear 2 Specifications.* Retrieved from `http://www.samsung.com/global/microsite/gear/gear2_specs.html` (Accessed on 28/05/2014)

Sanchez, C. A., & Goolsbee, J. Z. (2010). Character size and reading to remember from small displays . *Computers & Education*, *55*(3), 1056 - 1062.

Sandars, J., Homer, M., Pell, G., & Croker, T. (2008). Web 2.0 and social software: the medical student way of e-learning. *Medical Teacher*, *30*(3), 308-312.

Schlegelmilch, J. (2014, February). Spartan Daily. In San Jose State University (Ed.), (p. 3). San Jose State University, School of Journalism and Mass Communications.

Seppala, T., & Broens, R. (2013, February). *How New Smartwatches Could Revolutionize the Artefact Surroundings of People* (Tech. Rep.). The Research Institute of the Finnish Economy Lnnrotinkatu 4 B FIN-00120 Helsinki Finland: The Research Institute of the Finnish Economy.

Shanklin, W. (2013, November). *Review: Martian voice control smartwatch.* Retrieved from `http://www.gizmag.com/martian-watch-review/29834/` (Accessed on 28/05/2014)

Sheusi, J. C. (2013). *Android Application Development for Java Programmers.* Cengage Learning.

Smailagic, S. D., A. (2002). Application design for wearable and context-aware computers. *Pervasive Computing, IEEE*, *1*(4), 20–29.

Smith, N. (2013, December). Classic Project: Pulsar P1 & P2 Quartz Wristwatch. *Engineering & Technology*, *8*, 100-101(1).

Sony. (2014a). *Open smartwatch project.* Retrieved from `http://developer.sonymobile.com/services/open-smartwatch-project/` (Accessed on 27/05/2014)

Sony. (2014b). *Smartwatch2.* Retrieved from `http://www.sonymobile.com/global-en/products/accessories/smartwatch-2-sw2/specifications/#tabs` (Accessed on 24/05/2014)

Spritz. (2014). *Spritz reading.* Retrieved from `http://www.spritzinc.com/the-science/` (Accessed on 24/05/2014)

Starner, T. (2001). The challenges of wearable computing: Part 2. *IEEE Micro*, *21*(4), 54–67.

Starner, T. (2013). Wearable Computing: Through the Looking Glass. In *Proceedings of the 2013 International Symposium on Wearable Computers* (pp. 125–126). New York, NY, USA: ACM.

Sutherland, I. E. (1968). A Head-mounted Three Dimensional Display. In *Proceedings of the December 9-11, 1968, Fall Joint Computer Conference, Part I* (pp. 757–764). New York, NY, USA: ACM.

Swan, M. (2012). Sensor mania! the internet of things, wearable computing, objective metrics, and the quantified self 2.0. *Journal of Sensor and Actuator Networks*, *1*(3), 217–253.

Thorp, E. (1998, Oct). The invention of the first wearable computer. In *Wearable Computers, 1998. Digest of Papers. Second International Symposium on Wearable Computing* (p. 4-8).

Tizen. (2014). *Tizen os.* Retrieved from `https://www.tizen.org/` (Accessed on 29/05/2014)

Want, R., Hopper, A., Falcao, V., & Gibbons, J. (1992, January). The Active Badge Location System. *ACM Trans. Inf. Syst.*, *10*(1), 91–102.

Wear. (2014). *Android wear.* Retrieved from `http://developer.android.com/wear/index.html?utm_source=ausdroid.net` (Accessed on 28/05/2014)

Weyland, M. (2013). The internets common denominator. *Communication Systems VI*, *1*, 59.

Wobbrock, J. O. (2006). The future of mobile device research in HCI. In *CHI 2006 workshop proceedings: what is the next generation of human-computer interaction* (pp. 131–134).

Xiao, R., Laput, G., & Harrison, C. (2014). Expanding the Input Expressivity of Smartwatches with Mechanical Pan, Twist, Tilt and Click. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 193–196). New York, NY, USA: ACM.

Ye, H., Malu, M., Oh, U., & Findlater, L. (2014). Current and Future Mobile and Wearable Device Use by People With Visual Impairments. In *Chi 2014*.

Zieniewicz, M. J., Johnson, D. C., Wong, C., & Flatt, J. D. (2002). The evolution of army wearable computers. *IEEE Pervasive Computing*, *1*(4), 30–40.