A Modular Suite to Support Human-Smartphone Interaction for People with Motor Skill Impairments

Massimiliano Donati, Luca Fanucci, Fabrizio Iacopetti, Alessio Vecchio
Dip. di Ingegneria dell'Informazione - University of Pisa
Laboratorio Nazionale AsTech, CINI
Pisa, Italy
Email: m.donati@for.unipi.it, luca.fanucci@unipi.it, fabrizio.iacopetti@unipi.it, alessio.vecchio@unipi.it

***Abstract*—Mobile devices like smartphones and tablets are nowadays not only the de facto choice for interpersonal communication, social networking or Web access, but also an important tool for an increasing number of every-day activities. Thus, maximum accessibility for people with disability is very important to ensure social inclusion and to promote an independent style of life. In this paper we present a suite of hardware and software components that allows people with motor skills impairments to control a mobile device via switch sensors for the exploitation of specific user's residual abilities. A gateway and a headset adapter allow the user to connect switches or a power wheelchair control system to the smartphone in different ways (wired or wireless). A simplified user interface on the mobile device can be used to access the main functionalities (voice call, SMS, media player), whereas a scanning keyboard enables textual input in other pre-existing apps.**

I. Introduction

Mobile devices like smartphones and tablets represent the de facto choice for personal communication, social networking and general Web access. Moreover, the possibility of accessing information anytime and anywhere and the always increasing number of features (e.g. NFC and GPS) transformed these devices into almost-universal tools for carrying out every-day activities, such as mobile payment and navigation. Many of these functionalities are particularly important for people with disabilities as they provide the opportunity of increased autonomy, safety, and social inclusion [1], [2], [3].

Major manufacturers conceive their devices according to a "design for all" strategy, nevertheless the level of accessibility remains in some cases unsatisfactory. Moreover, to follow the requests of mass-market customers, devices are getting richer and richer in terms of functionalities, but at the same time also more complicated for impaired users. Specialized devices for people with disabilities are produced or customized by niche manufacturers, but they can be hard to find and expensive [4].

From the point of view of accessibility, one of the main problems of standard smartphones is represented by their complex user interfaces: GUIs are composed by many small widgets and input methods increasingly rely on soft-keyboards, multi-touch, or gestures. These mechanisms are particularly cumbersome for users with perceptual, motor, or cognitive disabilities, who may not be able to select an area of the screen with sufficient accuracy or with the requested timing. Obviously these difficulties, which increase for those users affected by multi-impairments [5], represent an obstacle to a complete social integration, and may reduce self-esteem, independence, and safety.

This paper presents a suite for enabling people with motor skill impairments to access autonomously the main functionalities of a standard smartphone. The suite includes both hardware and software elements, that can be composed according to the needs of the user. The suite is dedicated to Android-based smartphones and is compatible with a large fraction of currently available devices.

II. Related Work

The operating systems of modern mobile devices natively provide some configurable accessibility features, in particular for those users who have visual and motor skill impairments. For visually impaired people, accessibility functionalities include speech synthesis of displayed text, vocal description of the selected widget, increased contrast, and screen magnifiers. For upper limbs motor impaired people, accessibility features include the possibility to answer incoming calls easily (e.g. using the home physical button or just touching the screen, instead of dragging an icon), or the possibility to reach the most commonly used functions (e.g. settings, restart, etc.) by means of an on-screen draggable button.

Another feature of present-day devices, conceived for being used by everybody and possibly useful for people with upper limbs disabilities, is voice recognition. Such a functionality allows the user to place calls, read/compose messages, and browse the Web (with some limitations), but in general it does not allow her/him to control many other functionalities.

Nevertheless for many disabled users, mostly those affected by severe motor skill impairments, accessibility features provided by the operating system are not always satisfactory [6]. For example, as mentioned, when movements are imprecise it is still possible to answer a call just by touching the screen, but using the software keyboard remains a tricky task. Moreover, the voice recognition functionality relies on good pronunciation and scarcely tolerates strong accent, emotion, or temporary illness [7]. As a consequence, it is generally not suitable for those users who also have speech deficiencies, which is not uncommon when motor skill impairments are caused by neurodegenerative diseases like Duchenne Muscular Dystrophy, Amyotrophic Lateral Sclerosis, cerebral palsy, etc. Finally, it is worthwhile to highlight that the performance of voice recognition is reduced in noisy environments [8].

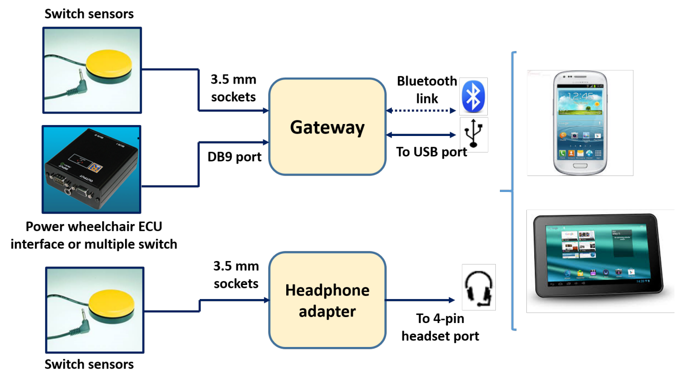

Fig. 1. System architecture.

Fig. 2. A prototype of the system: smartphone, gateway, and two switches (headset adapter not shown).

Specific sensors, acting as electrical switches, have been used in the last decades for the exploitation of residual physical abilities, in order to interact with Environmental Control Units (ECUs), infrared remote controls, and PCs. Examples of such switches include: pushbuttons with a large surface or reduced activation force, sip-puff and tongue-sensors, eye-blink detection sensors, sensors able to detect finger movements or other muscular activity [9]. Also the wheelchair-driving joystick, once properly configured, may act as a multiple switch sensor and it can be connected to external devices. These sensors operate in combination with opportune scanning control strategies. Scanning control consists in cyclical highlighting, one by one, all the items within a set (e.g. icons, buttons, menu items, etc.), whereas selection is triggered by the user's input. Various scanning strategies are possible, depending on user's abilities [10]: automatic scanning is time-based and requires a single switch sensor to select the currently highlighted item; manual scanning requires a second sensor to move the focus to the next item; other scanning strategies may use more sensors to highlight items along different directions.

To increase accessibility, some commercially available solutions have been recently presented [11], [12], [13]. Some of them are software-only systems whereas others also include hardware elements. Tecla and iPortal (dedicated to Apple devices) use the native graphical user interface, whereas input relies on external switches connected to a Bluetooth hardware device [11], [13]. ClickToPhone provides an ad-hoc and accessible GUI, which is mostly implemented using linear layouts and high visibility icons, that provides access to the main functionalities (e.g. phone calls) [12]. In general, a dedicated GUI specifically designed according to the users' needs is more effective, though it may not be able to provide access to all possible features.

III. A Modular Suite for Supporting People With Motor Skill Impairments

We present a suite for supporting people with motor skill impairments. The basic idea is to provide users the possibility of accessing the main smartphone functionalities using commercial switch sensors compatible with residual users' abilities (such as buddies, tongue-sensors, or eye-blink detection sensors) or custom keypads. Input can also be provided through integration of the smartphone with a power wheelchair. The suite also supports human-smartphone simplified interaction via a dedicated app. It also provides mechanisms for using the above mentioned physical input devices with existing apps. The suite is composed of four elements: i) a hardware device named gateway, ii) an adapter for connecting switch sensors, or any other switch, to the headset port iii) a portal app (ePhone), iv) a software keyboard. The system architecture is illustrated in Figure 1. A prototype of the system is shown in Figure 2.

The development of the system started from an analysis of users' requirements concerning the control of a mobile device in different use cases, i.e. a wheelchair user, a person lying in the bed, sitting at a desk, etc. During the development stages, some feedback from trial end users allowed tuning up some usability aspects of both hardware and software components.

A. The Gateway

The gateway is an electronic device that operates as an interface between the mobile device and switch sensors. On the user's side, it gets connected to one or two switch sensors through 3.5 mm mono jacks, which represent the de facto standard for disability specific switches. The gateway is then connected to the mobile device through a USB or Bluetooth link. The gateway prototype is shown in Figure 2.

The firmware on the gateway microcontroller detects the state of the switch sensors operated by the user and informs the mobile device of each switching event through data packets. Data are interpreted by the portal app and are translated into GUI commands, e.g. item selection, highlighting of the next item, etc. The gateway also provides a DB9 connector in order to manage a standard DB9 5-switch multiple interface and a 6-pushbutton custom keypad. Additionally, the DB9 port allows connection to the power wheelchair Environmental Control Unit port (ECU) to map the wheelchair-driving joystick positions to application commands.

When designing the gateway, the main requirements have been portability, small size, no need of external power supply, flexibility of use. As a compromise between compactness and battery life, the gateway is powered by 2 AA rechargeable batteries, which ensure a battery life of about one day for a typical use. Battery low condition is signalled by a dedicated LED. The size of the gateway is 90 mm x 65 mm x 16 mm.

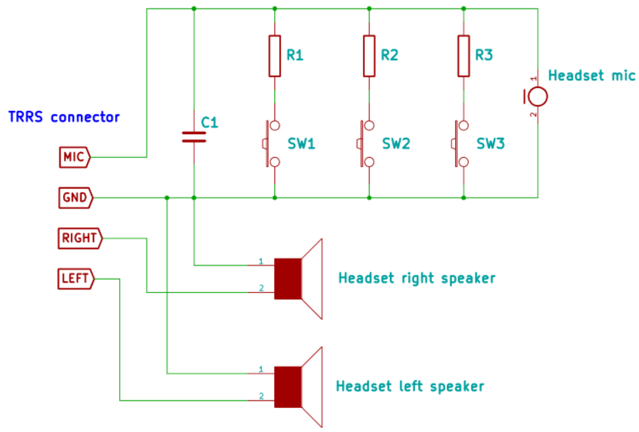

Fig. 3. Schematic of a headset with three switches.

B. The Headset Adapter

Depending on the user's lifestyle, the adoption of the gateway may be not strictly necessary. For instance, in some circumstances, the wireless connection between sensors and the controlled device may be less practical than a wired solution (wireless devices are battery operated). Similarly, if a power wheelchair is not used, some interfacing functionalities provided by the gateway are not needed. For this reason we devised a second simple possibility for connecting physical switches to the smartphone via its headset port. In fact, many mobile devices can be partially controlled using a set of switches embedded within commercially available headsets; such switches are commonly used to play/pause music, to select the next/previous track, or to answer an incoming call. We decided to exploit the same mechanisms for controlling the scanning procedure.

The smartphone headset port is generally a 4-pin port with the following signals: left audio out, right audio out, ground/common, microphone. The headset connector is normally a 3.5 mm (more rarely 2.5 mm) 4-pin jack connector (also called TRRS connector: Tip, Ring, Ring, Sleeve). A problem we found is the existence of at least two main reference standards for the implementation of the smartphone headset port, one from the Open Mobile Terminal Platform (OMTP), and another from the Cellular Telecommunications Industry Association (CTIA) standard in North America (the difference between OMTP and CTIA is an inverted mapping of the microphone and ground lines). Moreover, there is a lack of information about the internal circuitry of headsets produced by different vendors. The typical schematic of headsets is reported in Figure 3.

In Android smartphones, according to official specifications, support for CTIA pinout is required, whereas OMTP is optional [14]. However, we found that terminal producers do not strictly follow such recommendations.

Resistors R1, R2, and R3 have different values. When a given pushbutton is pressed, the impedance seen from the smartphone circuitry between the MIC and GND terminals is a function of the resistor value and the microphone impedance. Different impedance values are associated by the operating system to different headset events, typically *Hook*, *Volume* +, *Volume* - events, which are used to control phone calls, media player, and other applications. Android compatibility specifications indicate the impedance values corresponding to such events: 0 Ω, 240 Ω, and 470 Ω. Another event, corresponding to an impedance value equal to 135 Ω, can be recognized in some devices and it is generally associated to Google voice search.

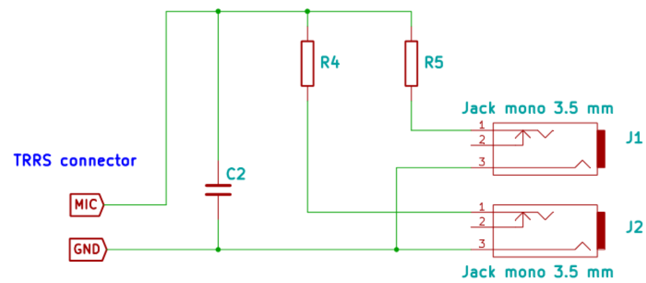

Fig. 4. Schematic of the headset adapter.

In order to apply the headset control mechanism to disability-specific switch sensors, we built a small adapter connected on one side to the headset port of the smartphone (according to CTIA specifications) by means of a male TRRS connector and on the other side to up to two physical buttons. The adapter is basically a set of RC networks and connectors and thus its cost is very limited, see Figure 4 and Figure 5. In particular, we built two adapters: the first maps the two physical switches onto *Hook* and *Volume* + events, the second onto *Hook* and *Volume* -. The adapters were tested and used to control the ePhone application and the scanning keyboard, where handset events are interpreted as scanning controls. They have also been tested with built-in Android 5.0 accessibility functions which enable scanning-based control of the smartphone. The adapters have been tested in combination with nine terminals, produced by four different vendors (LG, Alcatel, Nokia, and Samsung) and running Android versions from 4.2 to 5.0. Two devices provided an OMTP connector, and thus were not compatible with our implementation. For the remaining ones no significant problems have been experienced (with the exception of a single device, probably because supporting all the events is not mandatory according to Android specifications).

The positive aspects of this solution are: i) the adapter does not require batteries; ii) it is very cheap; iii) it can be compatible with all devices that are compliant with Android specifications; iv) it can be used with disability-specific switches already owned by users. On the other hand, the headset adapter cannot be connected to more than two or three physical switches for most Android smartphones, and is not suitable to both OMTP and CTIA pinouts, although a simple inversion of MIC and GND terminals in the TRRS connections should be sufficient (OMTP/CTIA adapters are anyway quite cheap, in the range of a few Euros).

The current implementation of the headset adapter does not allow the user to connect also a standard headset in addition to the switch sensors. This could be made possible by adding a TRRS socket to the adapter, even if in this case the effective resistance between MIC and GND terminals would be a function of the microphone impedance too. As specifications

Fig. 5. The headset adapter connected to two switch sensors.

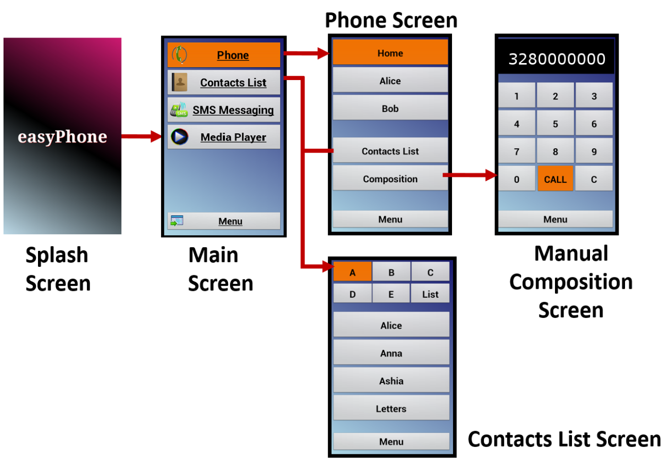

Fig. 6. The main screens of the portal app.

do not provide an exact value for the microphone impedance, but only the value of the impedance seen between MIC and GND terminals for each event, we expect that with headsets from different brands the resistors should be tuned to different values (with the exception of the Hook event). To this aim, in order to achieve a plug-and-play solution, a preliminary study has been carried out on the possibility of implementing an intelligent device based on a self-calibrating approach, where solid-state resistors can be programmed to the required values thanks to an app-device interaction (the app would command the device to simulate a switch press and to change the resistor values until events are correctly recognized). Moreover, such a solution could potentially be suitable for both OMTP and CTIA connectors.

C. The Portal App

The ePhone "portal" app offers a simplified access to telephony, contacts list navigation, SMS messaging and media player. Through this app, the user can select and interact with the mobile device by means of external switch sensors connected through the gateway or through the headset adapter.

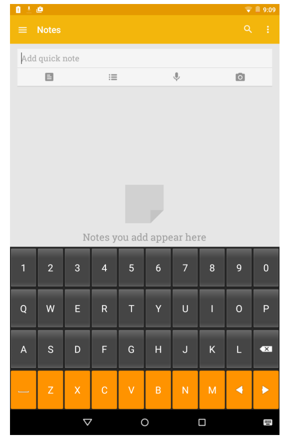

Fig. 7. The scanning keyboard can replace the standard Android keyboard.

The application can work in the following configurations: i) timed scanning and one switch sensor, ii) manual scanning and two switch sensors, iii) manual scanning and multi-directional navigation with a 5- or 6-multiswitch. Depending on the app settings, the touch screen may still be used as an additional input source, addressing the needs of some users with mild limb disabilities.

In order to improve accessibility for target users, the ePhone app replaces the screens offered by the corresponding native applications with screens based on ad-hoc layouts and contents. In details, layouts implement a linear placement of controls (buttons, icons, text, etc.), high color contrast, large icons, large fonts, a low number of nested screens, to reduce user effort and to help in case of mild visual impairment.

The ePhone app provides an initial configuration screen controllable with the native touch interface; access to such settings is not needed again unless the control strategies have to be changed. Settings mainly allow the user to select between gateway and headset adapter and to configure the scanning strategy.

The ePhone app is compatible with Android smartphones and tablets starting from version 2.3 when Bluetooth is used and version 3.0 when the gateway is connected via USB. Figure 6 shows the main screens of the ePhone portal app.

D. The Software Keyboard

To enable textual input in other applications, i.e. to access functionalities not supported by the portal app, the suite includes a software scanning keyboard that replaces the standard Android keyboard. Once installed, the new keyboard appears as one of the possible input methods within the Android settings. If selected as the default keyboard, every time the user is requested to input text, the scanning keyboard is displayed. The keyboard operates both with timed or manual scanning strategy, to be used respectively with one or two switches. Switches can be connected to the smartphone either using the gateway or using the headset port. Figure 7 shows the scanning keyboard used to insert text in an application.

IV. Discussion and Conclusion

Two of the most important requirements of a system for supporting the interaction between a motor skill impaired user and a mobile device are flexibility and configurability, as they enable fine-grained customization depending on the user's needs and conditions (e.g. user sitting on a power wheelchair, lying in the bed, sitting at a desk, etc.).

For this reason, we designed and implemented a suite of hardware/software components that can be organized according to the necessities of users. The gateway and the adapter can be used to connect one or more switches to the smartphone: the former provides more flexibility and does not requires wires, the latter has a low cost of production and is not battery-operated. The software components allow the smartphone to receive input from external buttons and provide simplified access to most commonly used functionalities (the portal app) or enable textual input to already existing applications (the scanning keyboard). In all cases, simplicity and reduced costs have been the design guidelines.

Despite its simplicity, the use of the headset port as a possibility for connecting switch sensors has not been previously considered (as far as we know). With respect to the other analyzed reference solutions, as discussed in [15], the ePhone software application also simplifies the strategy to access most common functionalities, which means a reduced number of elementary actions for the user and therefore reduced effort and time. The software keyboard operates as a standard Android input method and therefore it enables textual input in other apps. We believe that the modularity of our suite and its reduced cost make it a useful addition to the landscape of assistive solutions.

As far as the scanning keyboard is concerned, future improvements will include word prediction and customizable position of commonly used buttons (e.g. backspace, ".com", etc.). The different configuration of the system will be proposed to trial users in order to collect additional hints and feedback. The adapter is currently undergoing a studying phase considering the possibility to support also the connection and use of a generic headset. The software components will be available on Google Play for free.

References

[1] UN Broadband Commission for Digital Development, G3ICT, IDA, ITU, Microsoft, the Telecentre.org Foundation and UNESCO, "The ICT Opportunity for a Disability-Inclusive Development Framework," 2012.

[2] J. Abascal and A. Civit, "Mobile communication for people with disabilities and older people: New opportunities for autonomous life," *Proceedings of the 6th ERCIM Workshop*, pp. 255–268, 2000.

[3] S. Abbate, M. Avvenuti, F. Bonatesta, G. Cola, P. Corsini, and A. Vecchio, "A smartphone-based fall detection system," *Pervasive and Mobile Computing*, vol. 8, no. 6, pp. 883 – 899, 2012, special Issue on Pervasive Healthcare. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1574119212000983

[4] S. K. Kane, C. Jayant, J. O. Wobbrock, and R. E. Ladner, "Freedom to roam: A study of mobile device adoption and accessibility for people with visual and motor disabilities," in *Proceedings of the 11th International ACM SIGACCESS Conference on Computers and Accessibility*, ser. Assets '09. New York, NY, USA: ACM, 2009, pp. 115–122. [Online]. Available: http://doi.acm.org/10.1145/1639642.1639663

[5] H. Nicolau, J. Guerreiro, and T. Guerreiro, "Stressing the boundaries of mobile accessibility," *arXiv preprint arXiv:1402.1001*, 2013.

[6] P. Encarnacao, L. Azevedo, G. Gelderblom, A. Newell, and M. N., *Assistive Technology: From Research to Practice*. IOS Press, vol. 33.

[7] S. S. Kajarekar, H. Bratt, E. Shriberg, and R. de Leon, "A study of intentional voice modifications for evading automatic speaker recognition," in *Speaker and Language Recognition Workshop, 2006. IEEE Odyssey 2006: The*, June 2006, pp. 1–6.

[8] P. Jaffray, "Minneapolis street test of google s-voice vs ios siri," 2012.

[9] EASTIN - European Assistive Technology Information Network, http://www.eastin.eu/en/searches/products/list?iso=240918.

[10] A. Cook and J. Polgar, *Cook and Hussey's Assistive Technologies: Principles and Practice*. Mosby Elsevier, 2007.

[11] Komodo OpenLab, "Tecla," http://komodoopenlab.com/tecla.

[12] ClickToGo, "ClickToPhone," http://www.click2go.ie.

[13] Dynamic Control, "iPortal," http://www.dynamiccontrols.com/iportal/iportal-accessibility.

[14] Android, "Wired audio headset specification," http://source.android.com/accessories/headset-spec.html.

[15] M. Donati, F. Iacopetti, D. Mandoloni, F. Giometti, and L. Fanucci, "ePhone - a Technical Aid System to Ease Accessibility to Android Smartphones for Motor Skill Impaired Users," in *Proceedings of the 12th European AAATE Conference*, 2013.