Vision Based Interaction Techniques for Mobile Phones: Current Status and Future Directions

Simon B. Kerr, Greg Foster, and Hannah Slay

Abstract—In recent years mobile devices have been deployed with various new technologies, such as high quality cameras and the ability to support rich multimedia. Vision based technology such as motion detection has, until recently been limited to more powerful desktop devices. This paper lays out a brief review of these technologies with the aim of introducing the concept of vision based interaction on mobile devices and substantiate an implementation thereof. We conclude that mobile phones are currently being deployed with hardware and software which can support vision based interactions and which in the future could be widely deployed. We predict that in the near future vision based interactions such as gesture control will become prevalent and greatly enhance mobile devices.

Index Terms—computer vision, gesture tracking, human-computer interaction, mobile devices, mobile interaction.

I. INTRODUCTION

MOBILE devices have increased in both technology and popularity over the past few years [1]. However, many of the technologies available on mobile devices are not being used with secondary functions (i.e. providing a vision based interface) in mind. Mobile games for instance are restricted in their genre (i.e. arcade) and game depth by the simple keypad interface provided by most mobile devices [2]. The field of gesture control is fairly well established on desktop systems [3]. However, only recently, has the hardware of mobile devices allowed gesture control to be viably implemented in the mobile environment. The aim of this paper is to explore the progression of mobile camera based technologies over the past few years. We investigate the viability of using a vision based interface to control interactive applications such as mobile games and interactive tasks (e.g. navigation, turning pages, etc) without obscuring the phone's screen. This in turn has the potential to provide a much more usable interface to the user.

II. MOBILE DEVICE VISION BASED TECHNOLOGIES

This section explores the various fields in which recent technological advancements have been made on mobile phones. Five main areas will be discussed, the first of which reviews the various desktop and PDA vision based technologies. Secondly, a brief review of mobile device technological advances is presented. Thirdly, we explore image manipulation, which deals with editing pictures at the pixel level on mobile devices. Fourthly, motion detection is presented which explores the use of the camera on the mobile device to implement computer vision. Finally, the use of mobile devices as mobile interfaces for other devices is explored.

A. Desktop and PDA Technologies

Vision based interactions have been implemented in various forms on the desktop platform, [4]-[7] as well as Tablet PC stylus based gesture control [8]. Because the desktop environment offers the most powerful hardware it is naturally the starting point for most new hardware intensive research. Hence, vision based principles formed on the desktop system have been ported over to mobile devices [9]. Whilst the core principles stay the same between desktop and mobile device, the environment is different and hence the requirements for vision based interactions differ.

PDA's have long been the closest mobile device to the desktop platform and are equipped with the ability to use stylus based gesture control [2], [10], and [11]. Current PDA hybrids are equipped with cameras (Fig. 1b) and have seen significant growth in their hardware capabilities. Today the PDA is being replaced by the smart phone (Fig 1c) which combines the processing power of PDAs with the multimedia capabilities (i.e. camera) of high-end mobile phones.

[1] S.B. Kerr is with the Computer Science Department, Rhodes University, Grahamstown, South Africa (phone: 046 6038291; fax: 046 6361915; e-mail: g03k0704@campus.ru.ac.za)
G. Foster is with the Information Systems Department, Rhodes University, Grahamstown, South Africa (e-mail: g.foster@ru.ac.za).
H. Slay is with the Computer Science Department, Rhodes University, Grahamstown, South Africa (e-mail: H.Slay@ru.ac.za)

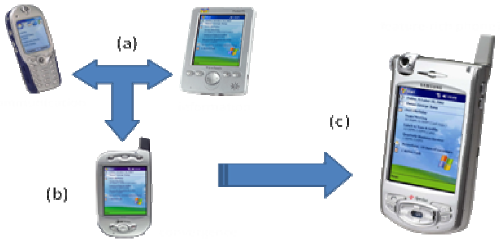

Fig. 1. Showing the combination of (a) mobile devices and personal digital assistants (PDA) called (b) XDA's. An amalgamation of all of these technologies is shown in (c) where both the task driven PDA and multimedia rich mobile phone are combined into one smart unit.

B. Mobile Device Technological Advances

Mobile phones have increased in core technology and branched into various specialized functions. Early mobile phones simply had the ability to place calls and send short text messages [12]. Displays were simple dot matrix screens, and these devices had slow processors and only a few kilobytes of memory (Fig 2a). Screen display quality of mobile phones has progressively increased both in their colour depth and screen size (Fig 2b), as has the memory capacity. Current phones have processors which reach speeds of 332Mhz (e.g. ARM 11 in the Nokia n95) and high resolution screens (Fig 2c) which allow high quality camera applications to be operated such as image manipulation applications [12]. There are now also gigabytes of space available to modern smart phones and up to 64 megabytes of RAM. These advancements allow vision based technologies to be developed and deployed on mobile devices which previously were not powerful enough or lacked embedded vision based technologies.

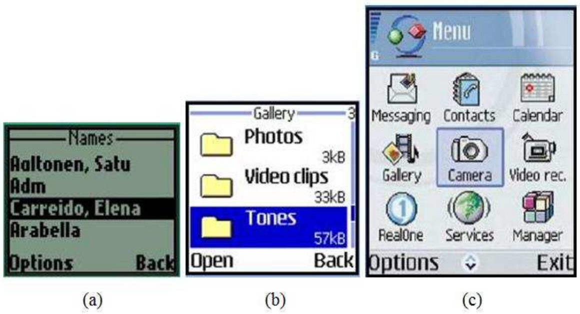

Fig. 2. An example of the progression of interface type and screen size in (a) Series 30 (84x48), (b) Series 40 (128x160) and (c) Series 60 (175x208) Nokia mobile phones.

C. Image Manipulation

Various image manipulation libraries and utilities have been developed over the past few years and are now being deployed to mobile devices. These include technologies such as *Search by Camera! ER Search* [13] which uses image recognition to compare images taken with mobile cameras and allow the user to obtain information on the subject of the photograph. A user takes a picture of a product (such as a wine label) and sends the image to a server which compares the image with pre-installed data and sends the user information of the product. Wilhelm *et al.* [14] use a guessing algorithm (using a repository of annotations) to help with photo annotation on a mobile device which allows for intelligent tags and input to be generated. These examples aim toward a more flexible mobile environment for the user. This flexibility is coupled with the ability to interact in a much more unique fashion based on the hardware which is currently deployed on mainstream devices now or in the near future. For example, the Nokia n96's high quality camera optics will allow a high quality picture to be modified using the Nokia Computer Vision library (discussed in following section).

D. Motion Detection

Various new technologies have become standard on current smart phones. Most notably accelerometers are now being deployed and smart phones such as the Nokia n95 and Apple i-phone now come equipped with this technology. Currently accelerometers are being used to orient the display according to the way in which the device is being held, but there are examples of accelerometers being used for other purposes, such as mobile games [15].

A recent study by Bucolo and Billinghurst [16], use a mobile phone's camera to control a handheld maze game. The phone is tilted to mimic the way in which a ball needs to be rolled in order to steer it around a maze. The camera interprets the angle by which the phone is being held based on contrasting objects below the camera. This is a highly complex activity which is implemented without the use of accelerometers. Whereas various vision based systems exist on computers, there are very few which have been implemented on mobile devices. This can be attributed to the fact that mobile hardware has only recently evolved to a point where a satisfactory blend of speed and accuracy can be achieved.

E. Mobile Interfaces

Mobile interfaces refer to the use of mobile devices to act as wireless interfaces or information gatherers for more cumbersome static devices (e.g. servers, vending machines and so forth). These devices either do not have an interface or they are not very user friendly. Sharp [17] maps out the use of mobile phones as portable interfaces for other devices. The use of the camera allows the mobile device to identify key features of the static device. The mobile device thus becomes an intelligent interfacing unit by using the camera as the main vision based interaction tool. The visual tag (developed by High Energy Magic) is read at 15 fps and highlighted on the device's screen using a crosshair. The device is able to identify and extract information using the camera and software deployed on the mobile device. This is a good example of a multimodal interface (the camera acts as a mouse which is coupled with the phone's buttons to act as mouse buttons) and displays one of the key future uses of mobile devices acting as mobile interfaces. A further example can be seen in Sharp, where a similar camera based recognition system was created [18]. This system reads visual tags via the camera, identifies them and then extracts the relevant information.

Increasing the power and variety of hardware on devices requires a similar increase in API's in order to obtain the best results. Therefore, many libraries and software packages have been released to support new hardware and take advantage of increasing hardware speeds.

III. MOBILE DEVICE SOFTWARE PROGRESSION

This section discusses the various platforms and software developments which have come to the fore. One of the most exciting for the fields mentioned in the previous section is the Nokia Computer Vision Library.

A. NokiaCV

The Nokia Computer Vision Library is a library created by Nokia to facilitate the lack of support for camera and image manipulation in the Series 60 SDK [19]. This platform is the most prevalent operating system used by current Nokia smart phones [12]. The Computer Vision Library provides key operations to process camera input, whether it is in video or static image form. Whilst, this library provides a convenient test bed to develop camera based interactions which will work on Nokia phones, it is however limited to the Nokia phone line-up and hence is limited in its deployability.

The library provides various classes for manipulating images at the pixel level which include both math and matrix representations. The move down to the pixel layer of image manipulation is very important as it allows for any type of operations to occur on images and videos. In addition, the speed of modern smart phones is now allowing these manipulations to take place in the mobile environment (which was limited by processing power in the past).

B. Java MIDP and MMAPI

One of the major disadvantages of the NokiaCV library is that the source code is not available to the public and only works on Nokia phones. The Java platform however, only requires the JVM to run and hence can run on any phone which has the JVM installed. This has led to Java being widely accepted on mobile devices as a development platform.

The latest version of the Mobile Information Device Profile (MIDP 2) introduces user interface and security enhancements, multimedia support, and RGB image data manipulation [12]. These enhancements allow developers to easily deploy interfaces, games and multimedia based applications to mobile devices. The MIDP runs on the Connected Limited Device Configuration (CLDC) which contains the virtual machine which the applications run on. Various optional API's can be run on top of the MIDP platform, such as the Mobile Media API (MMAPI).

The MMAPI enables MIDlets (deployed Java applications) to play and record audio and video data [20]. This can be coupled with the MIDP 2's ability to manipulate images at the pixel level and apply transformations to work with a series of images for motion detection and gesture control. This is a far step from the initial Java deployments on mobile devices which supported very limited multimedia interactions and were really only suited to 2D games and basic applications. Java now supports 3D games as well as various API's (such as the MMAPI) which allow a developer to deploy multiple multimodal based applications to mobile devices.

C. Microsoft .NET Compact Framework

The .NET Compact Framework (CF) was released in 2003 and has become the leading platform for Pocket PCs and PDA hybrids, running Windows Compact Edition (Windows CE) such as the HTC and Smartphone [21]. The CF is not very generic and this can be seen by the fact that the language runtime is embedded on devices in a memory foot print on a ROM chip. As with the NokiaCV library the applications created with the Compact Framework are limited and in this case will only deploy to devices which run Windows CE. The current release of the compact framework does not support pixel manipulation [22]. The next release of the Compact Framework (version 3) will include features which will better allow for motion detection and more usable interaction techniques, such as gesture control and pixel manipulation which are not catered for in the current release.

IV. HUMAN-COMPUTER INTERACTIONS

The human-computer interactions field of computing is vast and this paper will only explore a subset of the subject which relates to computer vision for gesture interaction. Gesture interaction allows the user to interact with more than one interface at the same time; hence, a look at multimodal interfaces follows.

A. Multimodal Interfaces

Multimodal interfaces process two or more combined user inputs, such as; gestures, speech, touch, head movements, key presses and so forth, in a coordinated manner with multimedia system output [3]. The advantage of multimodal systems can be described as "supporting more transparent, flexible, efficient and powerfully expressive means of human-computer interaction" [3]. In a study by Pirhonen *et al.* [2] which was conducted on gestures used on mobile devices, it was found that gestures could be used to increase a user's performance (in this case, using the touch screen on a PDA) when interacting with the device. This was attributed to the user not needing to focus on the screen while using the keypad. The user can simply gesture (e.g. draw a line from left to right) to interact with the device and not have to concentrate on using a keypad.

The concept of a multimodal interface should be at the fore when designing a new interface for advanced interactions and gaming. Because gaming is one of the most interactively intensive processes both to a device and user [23], a highly flexible and usable interface must be created in order to create a seamless environment for interaction. It is with this in mind that a brief exploration of gestures is presented in order to better understand the requirements of gesture control.

B. Gestures Relating to Finger Tracking

Whereas gesture recognition is not a new form of input for intelligent systems, it is still in very early stages of deployment on a large scale [24]. At their most basic level, gestures should be easy to understand by the user and the intelligent system for them to be successfully used and interpreted by both [25]. Various different types of gestures exist and are used by people everyday. It is however, currently not possible for an intelligent system to identify and understand gestures as people do. It is therefore important to strictly state the types of gestures which a device must recognize.

Mulder [26] lays out seven distinct features which must be taken into account when designing software for gesture recognition. These are laid out below and are defined in relation to creating a vision based finger tracking implementation on a mobile device (using a rear situated camera):

- Tracking Needs: The application requires fairly accurate and fast variable based input on the horizontal and vertical plane.
- Occlusion: Mobile device sensors are assumed to be on the back of the unit and hence the screen is not obscured.
- Size of Fingers: Fingers are held close to the camera and hence negate their small size.
- Context detection: Detection should occur when and if a finger is detected pointing upwards in front of the camera.
- Gesture Segmentation: Possible calibration ranges should be instantiated. Otherwise the edges of the screen can be set as the maximum values.
- Feature Description: As stated the hand is orientated facing away from the user with the index finger or thumb pointing upwards in relation to the camera (which is held above the finger).
- Gesture Identification: Gestures are interpreted based on the application which is running i.e. a racing game will treat the finger as a steering wheel, whereas a page turning application will turn a page when the finger passes a predefined maximum value.

As can be seen all of the requirements for an interaction system fit well into the above framework for gesture interaction. The problem of obscuring the screen is negated by the position of the camera. Fingers are fairly small and can easily be lost in the background [26]. This problem is rectified by the way in which the device is held close to the user's hand. The need for gesture recognition is also simplified by the fact that a single finger is capable of far less articulation than an entire hand. As new multimodal interfaces such as gesture control come to the fore, they must be coupled with applications which take advantage of this new interfacing technique.

V. APPLICATIONS

The mobile market is a very young and fast growing environment, Wisniewski *et al*. [15] have stated “strong growth and innovation in the wireless communications industry has led to the emergence of the wireless entertainment market.” Such factors as the rapidly increasing availability of multimedia phones, as well as the amount of subscribers and next-generation wireless networks can be attributed to this fast growth. Two main types of applications well suited for vision based interaction will be discussed in this section, mobile games and mobile applications, respectively.

A. Mobile Games

There have been some attempts in the mobile phone market to cater for gaming. The Nokia *Xpress-on* gaming cover provides the user with an extended keypad which makes interfacing with mobile games easier [27]. This however, is still an added extra which one must purchase to add to the gaming experience. Mobile devices tend to lend themselves more to casual gaming and many users will not buy additional hardware in order to increase the gaming potential of their devices. It is therefore up to the developer to make use of the current hardware to increase the potential of mobile interactive applications.

Currently the mobile gaming market is dominated by arcade and puzzle style games [15]. This may be attributed to the mobile interface (phone keypad) and the convenience of the type of game (casual game) and the amount of concentration required when playing games on the move. Games which are ported from other platforms to mobile devices and which bear the same name (e.g. Need for Speed, Grand Theft Auto, Call of Duty, Fifa, Halo 2 and so forth), are almost always changed in genre to suit the interface and environment (short gameplay on the move) provided by mobile phones. Realistic versions of games such as first person shooters are not suited to the current mobile phone environment as they require extensive multimodal input to provide the player with a realistic interface. A player must use a keyboard, mouse, sound and in a multiplayer environment even voice.

Anybody who owns a recently released mobile phone has the hardware to play mobile games. Many of these users are not going to spend extra money on such merchandise as the *Xpress-on* gaming cover or other types of peripheral hardware. Using the hardware already available as a fluid gaming interface is important for both the market and the user. Games which require pointing devices and a large screen will not work well on the currently available mobile platform [15]. We wish to highlight that this would be possible with vision based interactions.

Standard phones provide static keypads which can only detect single presses. At most, devices such as the *RIM Blackberry* provide scrollable functions, but this is not suited to the multitude of multimedia activities that require multi-value input.

B. Spatial Based Interfacing

Spatial based interfacing takes into account more than simple digital up or down input such as that represented in pressing a keypad key. There is no fuzzy area where more precise readings can be taken. Joysticks, steering wheels, analogue pads and the like all account for variance in user input (e.g. turn four degrees to the left). These types of analogue inputs can be used to more readily interpret what exactly the user is trying to input to the device.

The main downside of spatial based interfacing with games is that multiple key presses are not suited to the mobile platform. This technique of interfacing with games is prominent in arcade games where the user is required to either ‘button bash’ to achieve a result or to press a series of buttons in the correct order at a certain speed to attain the required goal. It is however unlikely that that these games work well on mobile devices as the keys on mobile phones are not designed to be pressed multiple times in succession at speed. If a spatial based interface is required but the hardware is not available, then certain solutions can be implemented, such as holding down a button to increase a value (e.g. hold down up to increase acceleration at a constant rate). This however is not very efficient when precise values are required at very precise intervals.

The games that are best suited to spatial based interfaces are any games that require fluid movement in three dimensions such as first person shooters or realistic racing games.

A genre of games which is well suited to gesture control is that of racing games. It should be noted that the majority of racing games require movement in only two dimensions with relation to the player (i.e. the player is only presented with controls to move forward, backwards, left and right). In arcade style racing games simple key presses are used to initiate slides and other racing manoeuvres which are sufficient to successfully control the vehicle through any given obstacles. However, realistic type games take into account real world physics and the corresponding action of simply pressing a button (turn steering wheel 100% right or left) clearly results in digital catastrophe with the player crashing. This is why a steering wheel or analogue gamepad is required for realistic simulations. Whereas a keypad can provide a good interface to a game designed for keypad interaction in two dimensions, it is however very limiting when moving to the third dimension (up and down). The movement values obtained from gesture control can be more likened to that of a steering wheel and hence most suited to that modality and those types of games. It can clearly be seen that there are very few if no realistic type games which are deployed on mobile devices. We predict that with the advent of vision based interactions, mobile devices will be able to provide an analogue interface for realistic type games.

B. Mobile Applications

Mobile applications form an integral part of the appeal of mobile devices. The advent of fully fledged internet access and QWERTY keyboards has allowed phones to perform almost all of the functions that a mobile computer can perform. Mobile applications range from word processors to web browsers. All of which are also limited to the layout of a mobile device's interface. Whereas QWERTY keyboards have allowed much more flexible character input mechanisms they still have the disadvantages of being small and increasing the size of the device. The use of a camera and a smart tag keyboard could solve the problem of typing on mobile devices. The camera could also be used for browsing where certain gestures could be used to scroll, turn pages or move pointers around. The authors predict that mobile applications can benefit greatly from a more robust interaction technique like vision based interaction.

VI. FUTURE DIRECTIONS

In this paper we have laid out the various hardware and software platforms which are implemented on mobile phones. Desktop and PDA based software has been ported to mobile devices to allow much more powerful applications to be deployed. The influence of more powerful hardware capabilities on mobile phones and their supporting API's leads us to predict that there will be a strong movement towards vision based interactions on these devices (Fig 3).

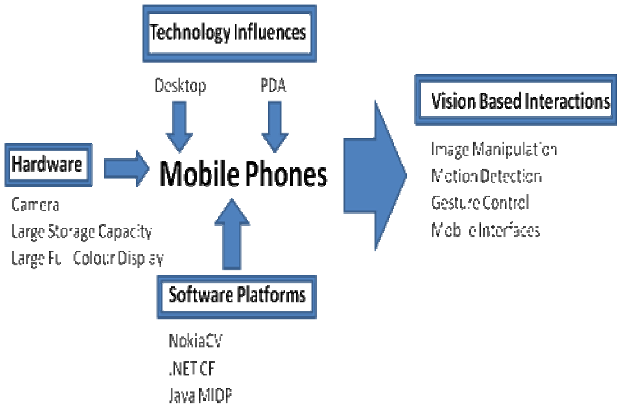

Fig. 3. Mobile phones are deployed with the hardware capabilities and software platforms required to implement and use vision based interactions.

VII. CONCLUSION

This paper has shown all of the recent developments on mobile devices which are related to vision based gesture interaction. The current strides in the underlying technology allow research into new human-computer interaction techniques, without additional hardware to take place. Creating multimodal interfaces with the use of gesture control will open many new possibilities for mobile applications. We predict that current trends and technological deployments lend themselves towards vision based interactions becoming standard on mobile phones.

The authors are currently investigating the possibility of creating a vision based system for finger interactions on current mobile phones. This involves creating a Java based MIDlet (MIDP 2.0 using the Java Wireless Toolkit 2.5.2) which can be deployed on various mobile phone platforms. The MIDlet accesses the camera and uses an efficient motion detection algorithm to identify and track the user's fingers. This information is relayed to an application which uses the gesture input to interact with the system.

REFERENCES

[1] G. Koprowski, Research: Mobile Phone Industry Poised for Spectacular Growth. *Tech News World.* [Online] 2007. Available: http://www.technewsworld.com/story/48423.html.

[2] A. Pirhonen, Stephen Brewster and Christopher Holguin. "Gestural and Audio Metaphors as a Means of Control for Mobile Devices". Minneapolis, *Conference Paper. CHI, 2002.*

[3] S. Oviatt, "Multimodal Interfaces". New Jersey : Lawrence Erlbaum, 2002. *Handbook of Human-Computer Interaction.*

[4] L. Bretzner, I. Laptev, and T. Lindeberg, "Hand Gesture Recognition using Multi-Scale Colour Features, Hierachical Models and Particle Filtering". *FGR '02, 2002.*

[5] J. Martin, V. Devin, and J. L. Crowley, "Active Hand Tracking". *GRAVIR - IMAG*, 1998.

[6] W. T. Freeman , K. Tanaka , J. Ohta,and K. K. Killington, "Computer Vision for Computer Games" : Mitsubishi Electric Research Lab, 1996. *IEEE 2nd Intl. Conf. on Automatic Face and Gesture Recognition.*

[7] J. Letessier, and F. Bérard, "Visual Tracking of Bare Fingers for Interactive Surfaces". Santa Fe, New Mexico : *UIST*, 2004.

[8] Microsoft. Windows XP. *What is a Tablet PC.* [Online] February 2005. Available: http://www.microsoft.com/windowsxp/tabletpc/evaluation/about.mspx.

[9] P. C. Arribas, F. Monasterio and H. Maciá, "Implementation of Camus Correlation Optical Flow Algorithm for Real Time Images". *FPGA* Madrid : Dpto. de Sistemas Electrò y de Control. E.U.I.T Telecommunicaciòn. Univ. Politécnica de Madrid, 2000.

[10] M. Lin, R. Goldman, K. J. Price, A. Sears and J. Jacko,"How do people tap when walking? An empirical investigation of nomadic data entry". Atlanta : Elsevier Ltd, 2007. *International Journal of Human-Computer Studies.* pp. Volume 65, Issue 9, Pages: 759 - 769.

[11] J. Stage and J. Kjeldskov, "New Techniques for usability evaluation of mobile systems". Elsevier Ltd, 2004. *International Journal of Human-Computer Studies.* pp.599 - 620, Volume 60, Issues 5-6,.

[12] C. P. Guan, *Developing MIDP 2.0 Applications For Nokia Devices.* [Online] 2004. Available: www.nokia.co.jp/forum/support/event/pdf/Nokia_Borcon2004_D2C4.pdf

[13] Research On Asia Group. *Camera Phone-based Mobile Search.* Tokyo : ROA Group, 2006. White Paper.

[14] A. Wilhelm, *et al."*Photo Annotation on a Camera Phone". Vienna , 2004. *Computer Human Interaction Conference.*

[15] D. Wisniewski, *et al. Mobile Games White Paper.* IGDA (www.igda.org/online), 2005. IGDA Online Games SIG Game Developers Conference.

[16] S. Bucolo and M. Billinghurst, "User experiences with mobile phone camera game interfaces". New York, ACM, 2007. 4th international conference on Mobile and ubiquitous multimedia.

[17] S. Richard, Overview: New Uses for Camera Phones. *Technology@Intel Magazine.* July 1, 2004.

[18] M. Rohs, B. Gfeller, and M. Friedemann. *Visual Code Recognition for Camera-Equipped Mobile Phones.* s.l. : ETH Zurich, 2006.

[19] NokiaResearchCenter. class_c_camus. *Nokia Research.* [Online] 2006. Available: http://research.nokia.com.

[20] J. Knudsen, Taking Pictures with MMAPI. *Sun Microsystems Development.* [Online] April 1, 2008. Available: http://developers/sun.com/mobility/midp/articles/picture/index.html.

[21] A. Wolfe, Microsoft's Compact Framework Targets Smart Devices. *ACM Queue.* 2003. Vol. 1, 7.

[22] Yakhnin, Alex. *How to Create a Microsoft .NET Compact Framework-based Image Button.* Windows Embedded Developer Center. [Online] March 2003. Available: http://msdn.microsoft.com/en-us/library/aa446518.aspx.

[23] B. Emily and P. Cairns, "A grounded investigation of game immersion".*Conference on Human Factors in Computing Systems: CHI '04 extended abstracts in Human factors in Computing Systems,* pp. 24-29, 2004.

[24] R. Sharma, V. I. Pavlovic, and T. S. Huang, "Toward multimodal human-computer interface". *Proceedings IEEE, 86[5](Special issue on Multimedia Signal Processing),* 1998.

[25] A. C. Long, Jr., J. A. Landay and L. A. Rowe. *Implications For a Gesture Design Tool.* University of California at Berkeley, 1999.

[26] A. Mulder, *Hand Centered Studies of Human Movement Project: Hand Gestures for HCI.* Simon Fraser University. s.l. : Simon Fraser University, 1996.

[27] Nokia. *Nokia leads GSM Evolution in Americas with new GPRS handsets*. [Online] March 18, 2002. Available: http://www.nokia.com/A4136002newsid=852457.

Simon B. Kerr completed his Bachelor of Science in 2006 majoring in Computer Science and Philosophy. He is currently reading towards a degree of Master of Science at Rhodes University.