A Device-Independent 3D User Interface for Mobile Phones Based on Motion and Tracking Techniques

J. Gimeno, P. Morillo, I. Coma and M. Fernández
Instituto de Robótica
Universidad de Valencia (Spain) *
E-mail: Jesus.Gimeno@uv.es

ABSTRACT

The user interaction in mobile phones has been traditionally based on key behaviors that users have adapted around their mobile devices. Although this type of interaction could be suitable for some contents, it could be improved from the point of view of the usability. In this paper we present a new user interface based on 3D navigation oriented to mobile phones. To enhance the user experience, we have developed a new interface based on 3D real-time graphics where users interact by means of natural movements of their devices. The goals of this development consist in enhancing the user interaction and accessibility to web content or interactive multimedia applications by means of avoiding a key-based, or a mouse-based, navigation and proposing a software solution adaptable to multiple and different mobile devices. In this sense, the user inputs can be detected by different input devices such as accelerometers or cameras, as well as the traditional keypads. Since mobile phones with on-board digital cameras are now widely available at low cost, the proposed 3D user interface exploits the acquisition capabilities of these input devices. In this sense, a differential algorithm has been applied in order to estimate phone movements from video images. The results of the performance evaluation of the 3D user interface shows that the proposed algorithm not only obtains a motion and tracking under extreme lighting conditions, but also adds an insignificant overhead to the system performance. Finally, a 3D environment has been designed to evaluate the performance of the presented approach, which has been successfully tested in actual users.

Keywords: Motion and tracking algorithms applications, Video-based capture, Mobile phones interfaces, 3D Navigation

Index Terms: I.4.9 [Image Processing and Computer Vision]: Applications—; I.5.4 [Pattern recognition]: Applications—[I.3.6]: Computer Graphics—Methodology and TechniquesInteraction techniques

1 INTRODUCTION

The improvements in the hardware of mobile phones have allowed running Virtual Reality (VR) and Augmented Reality (AR) applications on this type of devices. Nowadays, it is possible to overlay synthetic information on real images [10], or even to play 3D online games on PDAs, smartphones or some other mobile devices. However, the user interaction experience in mobile phones has been traditionally limited to the use of keypads or joystick controllers. The introduction of accelerometers, multi-touch screens and large-size displays has improved the user experience for the most recent mobile phones available on the market today. In this sense, the use of accelerometers in mobile phones has provided a suitable mechanism of interaction to navigate in virtual environments where intuitive interaction techniques are commonly demanded by users.

The integration of these new hardware capabilities into mobile phones has enabled to reproduce natural movements and inertias for mobile 3D real-time applications. Nevertheless, the use of accelerometers is still restricted to a small quantity of high-end mobile phones. In this sense, discrete keys are even sometimes the only available input device for some low-end cell phones available today.

In the present work, we propose a 3D interaction system that can use different input devices usually presented in mobile phones. The user movements can be obtained from the user-pressed keys, from the accelerometers (if they are available) or they can even be computed using video images captured by an on-board camera. Since the high availability of mobile phones including on-board cameras, the 3D user interface incorporates a motion and tracking mechanism based on a differential algorithm. This algorithm exploits the acquisition capabilities of these input devices to estimate the phone movements performed by the user. The proposed device-independent 3D interaction system has two goals. On the one hand, it makes possible to adapt the system to the hardware capabilities of the mobile phone. If the mobile phone is provided with an on-board camera or accelerometers, they will be used for interaction, but other mobile devices can take input events from keys. On the other hand, the system is used to navigate through interactive multimedia applications with a more natural interaction. For that, a 3D real time application consisting of a virtual environment has been designed. The aim is to simulate a virtual store where users gain access to different services offered by Orange Spain, a mobile phone operator. These services include live television, news, product offers and web access. The results obtained in the performance evaluation show that the presented approach can be validated as an efficient multi-platform 3D user interface for mobile phones.

The rest of the paper is organized as follows: Section 2 analyzes the related work on interaction techniques for mobiles devices as well as the weaknesses of the existing proposals. Section 3 describes the motivation and the aims for our approach. Next, Section 4 addresses the problem of developing a multi platform approach oriented to a wide variety of input devices for mobile phones. Section 5 describes the prototype application to test our navigation system, which is evaluated in Section 6. Finally, Section 7 presents some concluding remarks.

2 RELATED WORK

The improvement of the user interaction experience has been the purpose of different works and new interaction techniques for mobile devices has been proposed. In this sense, various technologies have been proposed and tested as input devices.

The introduction of touch sensors [8] provides the user a direct input from the screen by selecting options. Some mobiles devices use gesture recognition like HTC TouchFLO3D that detects four movements, or IPhone multi-touch screen that allows zooming and

*This work has been jointly supported by Orange R+D Spain and the Spanish MEC under grant DPI 2006-14908-C02-02

rotating images.

Other technologies like proximity sensors and accelerometers [10] have been tested. Hinckley uses accelerometers [8] for portrait and landscape display mode detection, and Wigdor [21] uses them for entering text and scrolling through documents. However, the main disadvantage of these technologies is that they only can be found in some expensive phones nowadays.

Camera phone has been the technology used by many researchers, due to the high quantity of mobile phones that includes it. The development of different algorithms and techniques for movement detection based on camera images has made it possible new interaction techniques.

Several researchers and companies use tracking systems to recognize visual barcodes with camera phones [13, 15]. These codes can be associated with different actions. Rohs [14] and Wagner [18] develop systems for the detection of 2D visual codes. Mohring [12] proposes a tracking solution for mobile phones that tracks color-coded 3D marker shapes. These solutions are mainly used in Augmented Reality systems for cell-phones where a 3D geometry is overlaid on live video stream. Although the value obtained with visual markers allows detecting absolute movements, in our work only relative movements are needed. Moreover, the necessity of markers is a constraint in the use of these systems.

A different approach to develop tracking systems is the detection of relative movements without markers. In order to achieve that some researchers use featured based algorithms. These algorithms tracks features like an edge o corner between frames to detect the direction and magnitude of movement.

Haro [3, 7] uses an algorithm based on tracking individual corner-like features observed in the incoming camera frames. Different applications are used to test the algorithm including zooming photo browser, document viewer a 2D game and a 3D game. For the 3D game interaction Quake III is loaded, but textures, light maps, curved surfaces and lighting calculations are disabled obtaining a rate of 3-10 frames per second.

Hannuksela [6] uses a Kalman-based tracker using efficient techniques for feature selection and feature motion analysis. Another A.R. game application has been made to play tennis using a color based feature detection method [5].

Hwang [9] also applies relative motion tracking to infer the motion of the user and this tracker is used to navigate through a virtual environment and to select objects. However, this algorithm is not evaluated in a phone mobile but in a hand-held PC model VGN-U71OP Sony is used.

These feature-based systems have a main disadvantage; they do not work properly with severe lighting differences, shadows or plain environments without features. Furthermore, this technique has a high computational cost due to the feature detection algorithm.

Another approach marker-less is the use of differential algorithms based on image differencing and correlation of blocks for motion estimation, similar to those used in MPEG2 and MPEG4 video encoders. According to Wang [19, 20] Tinymotion differential algorithm consists of four steps: color space conversion, grid sampling; motion estimation and post processing [19]. This software has been evaluated as a device for pointing, menu selection, text entry and playing well-known 2D games like Tetris.

Comparing the different detection systems, we have considered the use of a differential algorithm on the basis of two main reasons. On one hand marker tracking is not adequate for applications where environment cannot be restricted to a certain location and in the other hand differential algorithms work better than featured based with plain images.

Nevertheless, as we will explain in section 4, we have found that color space conversion to grey scale can be a problem with automatic white balancing of cameras, so we have use binaries images.

3 MOTIVATION

Orange Spain Mobile (belonging to the France Telecom Group) is the third largest mobile operator in Spain where serves more than ten million mobile phone customers, as well as a million Internet customers. Although one of the main activities of Orange consists in providing Internet access to general and business customers, this telecommunication company tends to launch a new business market based on maintaining multimedia 3G services. Since the use of 3G services is charged to customers according to the volume of data downloaded from the mobile phones, Orange maintains a broad set of 3G multimedia and entertainment applications to stimulate the customers to spend more time on the phone and, therefore, to boost their 3G data usage. Examples of 3G applications proposed by Orange, just like other telecommunication companies do, include video conferencing, electronic payment, on-line gaming, geolocation and precise positioning capabilities.

Unlike other countries that have launched 3G mobiles services, such as USA, UK, Korea or Japan, where 80 per cent of cellular users are subscribed to a mobile Internet service provider, the Western European customers need to be more motivated to assume an additional charge in the mobile phone bill for 3G data traffic. In order to attract customers and motivate them to use their mobile phones to access the 3G network, an eye-catching adaptation of the classic user interface provided by a mobile phone, to navigate through web content, could be performed. This adaptation should take into account the wide diversity of software and hardware platforms included in the most recent high-end mobile phones.

The purpose of this work is to develop and evaluate a multiplatform access portal to web content based on including both 3D real-time graphics and a new 3D user interface. The goals of this development consist in enhancing the user interaction and accessibility to web content by means of avoiding a key-based, or a mouse-based, navigation and proposing a software solution adaptable to multiple different mobile devices. In this sense, the software solution should allow users to navigate through websites and to handle multimedia content regardless the hardware features, such as accelerometers or cameras, which are often included in most of the current high-end mobile devices.

4 DESIGNING A DEVICE INDEPENDENT 3D-USER INTERACTION

A challenge of the present work has been to develop a 3D interaction technique adaptable to different mobile devices. We have implemented a navigation system to be used with three different input devices: keypads, accelerometers and on-board video cameras. The hardware devices available in a mobile phone will determinate the input device selected for the user interaction. Although accelerometers and on-board cameras will provide a more natural interaction than keypads, and therefore a more enhanced user experience, keypads will be always available in the whole mobile phone designs.

One of the main features in the initial design of the proposed 3D-user interaction was the capability of the software solution to offer an easy-to-use library when the interface is integrated into existing mobile phone applications. Moreover, it offers to the final mobile phone users a transparent and multifunctional interface, which are not usually common for mobile phone environment. In both cases, the work effort performed by mobile phone designers, and final users, is small since the proposed interface is very intuitive and does not require any type of reconfiguration or complex installation process.

For that purpose, we have designed a device-independent 3D interaction mechanisms acting in the same way over the three different input devices.

4.1 Key-based Interaction

Accelerometers and on-board cameras could be used to convert the user movements to regular input events. The user events are interpreted by the mobile phone applications to move the user through the 3D virtual world. This kind of interaction is more natural than key-based interaction mechanisms. Nevertheless, accelerometers and cameras are not always included in all the mobile phones available in the market today. Therefore, it is necessary to develop a system adaptable to mobile devices that do not incorporate these technologies.

The proposed key-based interaction is developed on the Joy-Pad of the mobile phone which is used to obtain four discrete movements: 'Up', 'Down', 'Right' and 'Left'. 'Left' and 'Right' movements are used to rotate user's point of view within the 3D virtual environment. 'Up-key' has two different uses. When Up-key is pressed shortly the current user's point of view is moved forward. However, when this key is pressed long, the interface shows a programable contextual menu. Finally, 'Down' movement is used to move the user backward.

In order to show the user a 3D interface regardless of the input device, both the accelerometer-based and the camera-based interactions follow the same criteria as key-based interaction, where the user actions obtained by the hardware devices are converted to a set of common input events. This criteria allows adding more hardware devices as input devices to the proposed interface as they become more available in the mobile phone market such as multi-touch screens, magnetic compass, etc.

4.2 Accelerometer-based Interaction

Nowadays, some high-end mobile phones include accelerometers that are used as tilt sensors. These sensors give the mobile orientation that is mainly used to change between portrait and landscape display visualization.

These sensors are used in our approach to detect the four movements needed: 'Up', 'Down', 'Right' and 'Left'. In order to obtain these movements it has been necessary to filter the data received from the three-axis accelerations.

If the original data received from accelerometers were used without filtering, any movement made by users would be transformed into movement in the virtual world. As a result, a slight involuntary movement, which are very common in devices hold by one hand, could be converted into unexpected displacements within the 3D virtual scene. To avoid that unpleasant behavior, the filter includes a threshold representing when the presence or the absence of movement is detected. In this sense, if the signal value from accelerometers in a given direction is higher than the predetermined threshold, then the movement is processed as a key-down event. This threshold has been empirically calibrated and validated on preliminary experiments by testing different values with actual users of the mobile application.

Moreover, the proposed data filtering provides two noticeable features since the 3D real-time application do not need the absolute position of the mobile device. On the one side, when a movement has been found and filtered, then the following movements are not processed assuming they correspond to the return movement of the device to the user's initial position. On the other side, the proposed data filtering eliminates one of the main problems that arise when accelerometers are used as motion capture device: "the drift error" [11]. When accelerometers are used to estimate the absolute positions of objects in motion, the acceleration signal output by the sensor is doubly integrated with time obtaining the covered distance. The offset exhibited in the acceleration signal is accumulative and the accuracy of the distance measurement deteriorates with time due to the integration.

4.3 Camera-based Interaction Using a Differential Algorithm

Although accelerometers are the most widely used devices to obtain the positions of moving objects in mobile computing, only few high-end mobile phones are equipped with them. As described above, keys can be also used as input devices, but an interaction based on movements of the mobile phone seems to be more natural to navigate and browse in 2D/3D virtual worlds. For this reason, we have implemented a third navigation system based on computing the user's movements by means of the visual information provided by the on-board camera included in most of current cell phones.

In order to estimate the hand gestures of the mobile device users from the images captured by the on board camera, we have developed a differential algorithm. Differential algorithms are currently used in a wide variety of image processing problems when images are sampled at a high rate [2, 16]. Basically, they determine the displacement of objects on consecutive images and are considered a better alternative than both marker-based or feature-based detection mechanisms for tracking systems in mobile computing [3]. The problem of tracking systems based on visual markers is that they are totally oriented to detect the position and orientation of physical elements (markers), which cannot be placed, left or glued in several actual applications [13]. However, the marker-less tracking systems, such as the detection systems based on visual features, avoid this problem, but require high computational power, which is not yet available in most of commercial mobile devices [17].

Although other mobile phone applications, such as TinyMotion [19], use basic implementations of differential algorithms to solve similar problems, we have had to introduce some modifications oriented to solve particular problems regarding the image acquisition, as well as to improve the global performance of the system. In this sense, the presented approach consists of the following steps: binarization, grid sampling and motion estimation. In all the steps, we have avoided to define a post-processing method, which requires more computing time, since we do not need to use absolute measures of movement from an initial position of reference.

4.3.1 Image Binarization.

The first step of our camera-based tracking system corresponds to an image binarization process where the original image, captured by the onboard camera, is converted to a binary representation. The purpose of the binarization process is to transform the enhanced color image into a black and white image for subsequent analysis and processing [16].

Unlike other approaches, such as TinyMotion [19], which proposes a color space conversion, we have decided to use a classic image binarization. The first step of the tracking technique included in TinyMotion corresponds to a color space conversion based on a bit shifting method where the captured 24-bit RGB color image is converted to an 8-bit gray scale representation. After implementing and testing the color space conversion technique on preliminary experiments, we found a serious problem related to the lighting images captured by the on-board camera.The camera phones manufactured today include an automatic white balance control to adjust the image such that it looks as if it is taken under canonical light. Since this automatic white balance control cannot be disabled in most of mobile phones, with very few exceptions, the gray level of the same captured object changes as the lighting condition of the environment varies. These unexpected variations of the gray level images for the same object cause detection errors, and therefore tracking errors.

In our case, we have solved this problem problem by binarizing the captured images using a dynamic threshold. This non-fixed value is computed in real-time taking the average and variance values of the pixel color of the capture image. Following this process, if an image changes as a result of a white balance process, the threshold also changes, and therefore the histogram remains almost in the same position indicating a highly correlated behavior. Additionally, this image binarization process consumes less memory than the color space conversion since data structure to represent a pixel corresponds to a bit (instead of a byte [19]).

4.3.2 Grid Sampling

Grid sampling is a well-known sampling technique very common in the field of the multiresolution image processing [16]. Basically, grid sampling sequentially converts the original pixel of the image into picture elements following a square pattern, denoted as sampling window. This technique is useful to reduce the computational complexity and the memory requirements when an image flow has to be processed in real time.

In our case, the pixels of the binarized image in this sampling window of the size 5x5 are transformed by means of a common average function. The size of the sampling window has been tuned to optimize the performance of the mobile application after much experimentation on preliminary tests.

Figure 1 shows the result of the binarization and the grid sampling process using a Nokia N95 cell phone as hardware test platform. Since this cell phone allows selecting some different video formats as image flow for a custom mobile application, we have selected the 128x96 resolution. The Nokia N95, which includes a 5-megapixel camera, produces a image flow consisting of 30 fps when this video resolution is selected. In our case, this video resolution has been chosen as a as an agreement between system performance and quality image. This figure shows that both processes acquire and process the images at a rate of 30 fps on this cell phone where the tuned sampling window converts the captured image into a 25x20 grid. In this example, the elements corresponding to the resulting binary matrix has been represented as red dots (values above the dynamic threshold) or white dots (values below the dynamic threshold).

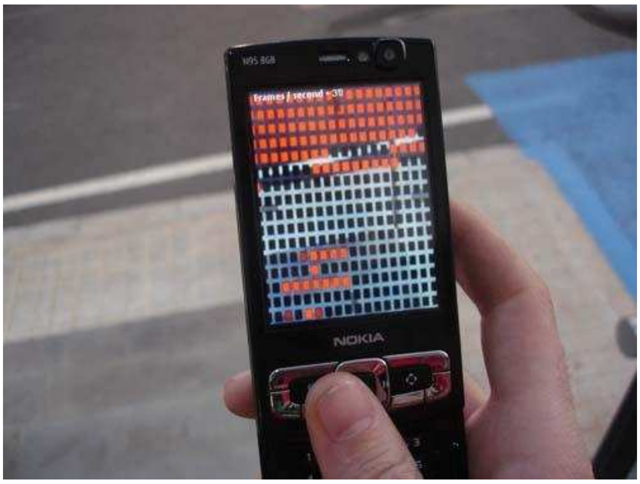

Figure 1: An example of the binarization and grid sampling processing running on a Nokia N95 at 31 fps

4.3.3 Motion Estimation

The motion estimation step estimates the final relative camera motion between these two frames since the user's mobile phone moves faster than the object located within the real scene. In our case, the motion result corresponds to the proper detection of any of the four moving directions: leftward, rightward, forward, and backward.

The motion estimation process takes a 25x20 matrix as data input, which elements are denoted as mega-blocks and were obtained from the original color image in previous steps. The size of this matrix is related to the performance of the selected mobile hardware platform, but it is invariant to the behavior of the algorithm. This process compares two consecutive frames shifting pixels in the four moving directions. This comparison is based on a pixel-to-pixel XOR operation of the two consecutive frames, where the separation of the pixels included in the XOR operations is increased until five times for each of the four moving directions. Each of the twenty XOR resulting matrixes represents the similarity of the consecutive images, which sum of the elements models the distance among them. We have denoted this value as distance error. For this reason, our motion estimation process selects as output the moving direction (among the four different alternatives) that contains the lowest distance error.

Unlike other approaches [19], our accelerometer-based interaction for mobile computing is based on actual binarization processes. The processes executes bit-level operations, which are much faster than byte-level operations. In our case, the proposed detection algorithm works on a search depth of 5-pixel distance, which allows detecting movements of the mobile device very fast. The byte-level operations support gray scale representations and require high computational power. This high computational power limits the search iterations, which prevents other approaches from exceeding a search depth of 3-pixel distance.

5 Experiments

We have implemented a prototype application to test our navigation system with different input devices. For this purpose, we have developed a mobile application called "3D Orange World". The aim of this test application is to simulate a virtual store where users gain access to different services offered by the leading telecommunication company Orange.

In this test application, a user can navigate through the virtual store by moving his/her mobile phone, PDA, etc. Initially, 3D Orange World is composed of three virtual rooms offering three kind of on-line services: Web-TV, newspaper service and advertising zone. The number of rooms in the virtual store, as well as the type of services offered, can be configured by the user with great flexibility. When a user selects a room, the mobile application simulates how the user moves and enters the chosen room in a 3D environment based.

In the television room the users can select and access to one of the several broadcast TV companies showed. The news room is represented by a newspaper kiosk and gives access to main newspaper websites. Finally, another room represents the Orange store where advertisements of different products are offered. Figure 2 show the execution of the 3D Orange World application on a Nokia N95 cell phone. In this case, the user is located in the entrance of the default configuration of the virtual store composed of the rooms.

The real-time 3D graphics application consists of the following modules: a 3D environment, a parser for the 3D object files and a 3D viewer. In order to simplify the designing process of the virtual world we have chosen a commercial tool. Thus, the 3D objects included in the 3D virtual scene have been designed using 3D Studio Max Autodesk tool. Since the 3DS files generated by this modeling tool cannot be loaded directly by the OpenGL library, it has been necessary to create a parser to included the 3D objects within the application.

The 3D viewer processes the virtual objects from the point of view of the user and renders a 2D image in the mobile screen. The 3D viewer has been developed using the standard OpenGL ES (OpenGL for Embebed Systems). Although others graphic libraries such as Performer, OpenSG or OSG provide high-level graphic data structures, called scene graphs [1], these libraries are currently not available to execute 3D real time applications on mobile phone hardware.

Figure 2: 3D Orange World displayed in a Nokia N95

In order to improve the performance of the rendering process of the mobile application we have included our own high-level library [4], which includes a custom scene graph and a culling process. This scene graph creates an internal structure where objects can be grouped depending on their locations within the 3D virtual scene. The graph enables an efficient visualization by clipping hidden objects from the user's point of view.

6 Evaluation

In order to evaluate the quality and limitations of the proposed 3D user interface based on motion and tracking techniques, a performance study of the developed application has been undertaken. This performance evaluation has focused on two main aspects. Firstly, the system performance has been evaluated by measuring the frame rate of the 3D application with the different input devices activated. Secondly, we have measured the human performance of the system by testing it with users. In particular, we have compared the three input techniques used as a 3D navigation and selection system.

The system has been compiled, deployed, executed and tested in two mobile platforms, which have been used for evaluation purposes. Nowadays, both hardware test platforms, corresponding to Iphone and Nokia N95 cell phones, represent two of the most important cell-phone available in the telecommunication market. The Nokia N95 mobile phone has a ARM-11 processor, 160 MB RAM and Symbian S60 3rd Edition Feature Pack 1 as operative system. Unlike Nokia N95, the Iphone includes a built-in three axis accelerometer, which can be controlled by the application developers using a commercial SDK provided by Apple. The IPhone has an ARM SoC-1176 processor, 128 MB RAM, a PowerVR MBX 3D graphic processor and a IPhone OS 2.0 beta3 5A240 operative system.

Figure 3 shows different images of the usability and final aspect of the developed test application running in both hardware test platforms. This figure shows how the automatic display mode, included within both mobile phones, allows the application to be executed in portrait and landscape orientations.

6.1 Performance Evaluation

A challenge of our approach was to provide the visualization of common 3D virtual environments in mobile devices at an acceptable frame rate and performance levels. Moreover, this achieved frame rate should be keep above a threshold when the 3D user interface based on video image processing was activated.

Figure 3: Different images of the 3D environment and the user interface with Nokia N95 and Iphone

The 3D virtual scene included within the 3D Orange World application consists of 22.000 vertexes. Nokia N95 can run 26 fps when this virtual scene is visualized using textures at the highest display resolution of the mobile phone. In the case of Iphone, this system throughput corresponds to 22 fps.

When the on-board camera is enabled and the proposed 3D user interfaces is activated, the actual SDK provided by Apple limits the rate of the image flow processed by the mobile phone to 5 fps. This is the reason why this mobile phone has been rejected to evaluate the performance of our 3D user interface based on video image processing. However, the Nokia N95 obtains an image flow consisting of 30 fps which, in turn, is processed and returned at a rate of 30 fps. Unlike other approaches [19], where the initial image flow consisting of 15 fps are reduced to a final processed images at a rate of 12 fps, these results show that the overhead of our proposed interface based on video image does not add a significant overhead to the system throughput. If both the visualization system and the 3D user interface are executed simultaneously, then the final system throughput corresponds to 15 fps, which could be considered as an appropriate threshold for mobile interactive multimedia applications [19]. Nevertheless, frame rate is not decreased when accelerometers or keys are used as input devices for this mobile phone.

6.2 User Evaluation

6.2.1 Experimental design

The experiment with users has been planned following the next steps:

System overview and free navigation. In the first stage of the process a brief overview of the experiments as well as the system purpose has been given to the participants. A short explanation of the 3D environment has been also provided, showing to user the contents of the rooms: news, television and orange shop. Finally, the interaction with the system using the different input devices has been showed. Once a participant has understood the system and the 3D proposed interface they have enjoyed a five minute free navigation to familiarize them with the interaction.

Tasks. To measure the system performance we have asked users for develop several tasks. Stage one. Firstly, a motion detection algorithm has been assessed and two motion tests have been performed by each participant. In the first test, the participants must complete 3 series of 10 movements using natural movements detected by the camera. The movement direction is displayed by the user by means of graphical arrows changing their color when a movement is detected. Total time spent in each group of correct movements has been stored. A second test was designed to quantify error rate of our camera detection algorithm. In this experiment, participants are asked for doing a group of 50 movements and the number of correct and incorrect movements has been collected. We compute the number of incorrect movements that cannot be completed at the first attempt.

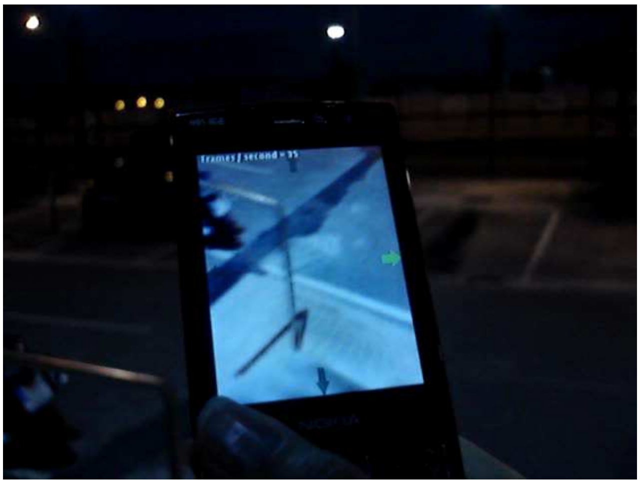

Figure 4: Evaluating the motion detection algorithm.

Stage two. The following stage of the test has been focused on user navigation through the environment and selection of multimedia contents. The participant were asked to find and select 8 contents located in different rooms of the environment using the three techniques of interaction (keys, camera and accelerometers) with the aim of compare them. During this stage, the executed time in each individual as well as main user problems has been stored.

User survey. After all the process, feedback from users has been collected by means of a survey. The user completed a questionnaire with general questions the use of mobile devices and questions about the application tested.

6.2.2 Experimental design

The experiment with users has been planned following the next steps:

The evaluation was made with 16 participants, 6 female and 10 male. All of them owned a mobile phone, and 6 of them a PDA. Some of the participants who have tested the system were selected among the members of the work group and thus we made differences between expert and novel users. The expert users are used to perform movements with camera and keys, while novels are only used to interact with mobile phone by means of the available keypad.

Regarding the user phones, 15 of 16 were provided with camera phones and 4 of them with accelerometers (IPhone and HTC Diamond phones). Finally, 40 percent of the users were accustomed to using Internet and 3D games with their mobile devices while 60 percent of them used sometimes, or never, these applications.

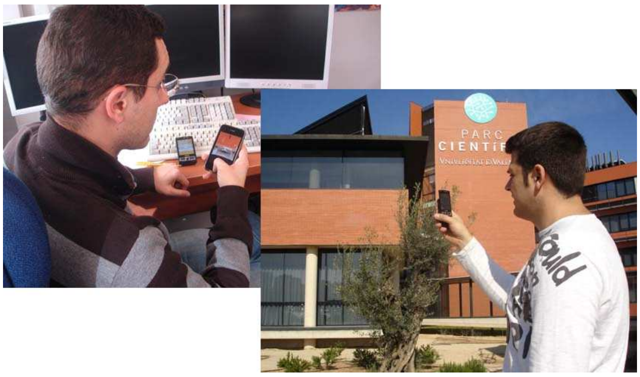

Figure 5: Evaluation test in different backgrounds

6.2.3 Results

In the first stage of the test a total amount of 1280 input movements made with camera were recorded. Differencing between expert and novels users the error rate is 4.6% for experts, and 8.4% for novels. It is necessary to remark that we have considered as fail trail each one not achieved at the first attempt.

Differencing the rate error between the four moving directions tested (up, down, left and right), we find a distribution of 30%, 27%, 17% and 26% respectively.

In the three groups of 10 movements we have found that users increase their speed to achieve the task. Figure 6 show how the time spent is decreased.

Regarding the tasks in the second stage of the evaluation, we have compared the time spent in each search task using the three input devices (keys, accelerometers and camera). Figure 7 show how total spent time is lower with keys and higher with camera.

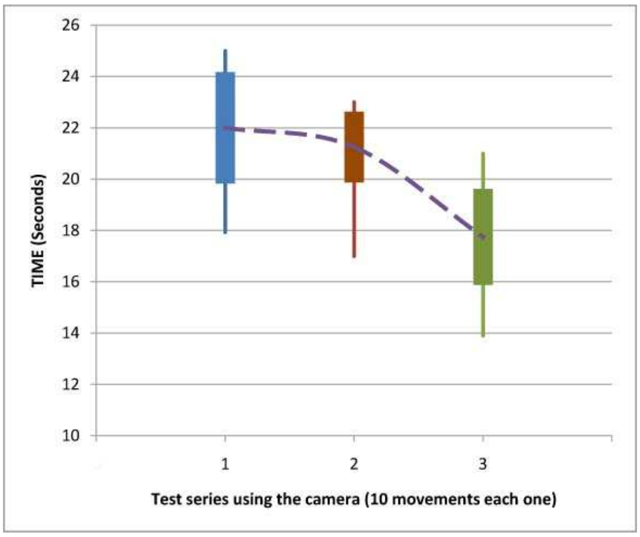

Figure 6: Application execution time for the three different groups of users

The different between expert and novel users, using the keys is very small. To perform the eighth task the experts users spent 80 seconds, while the novel users spend 60 seconds to accomplish the same tasks. This small difference shows that the novel users learned

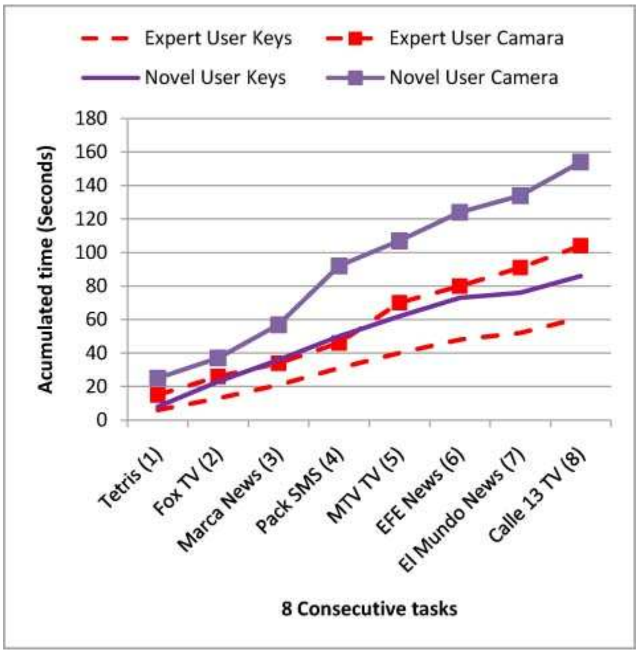

Figure 7: Results differencing experts and novel users in key-based or camera-based interfaces

very quickly the organization of the information and how to reach it using the key-based interface. The accelerometer time differences were higher than the differences produced by the key-base interface. However, the highest difference is found when the camera based method is used. In this case, expert users, who knew previously how this system and interaction method work spent 100 seconds performing the 8 task, while it took 158 to the novel ones. This higher differences when the camera is used, show that the learning process of the camera input method is the most difficult method, while novel users learned very fast how to use the key and accelerometer input methods.

7 Conclusions

In this paper we have proposed a device-independent system for 3D navigation on mobile phones. The interaction can be done using three different techniques as input devices: keys, video images captured by the camera or movements provided by accelerometers. The use of different techniques with the same 3D interface and interaction mechanism allows the adaptation of the system to the capabilities of the mobile phone.

The use of camera phone as input device has been implemented adapting a differential algorithm for motion detection of video images captured by phone-camera. We have validated the efficiency of our algorithm in different conditions.

Moreover, a challenge of the system has been accomplished, and we can state that the system works properly displaying the 3D environment at 22 frame rate while the motion detection algorithm is computing images.

References

[1] K. Bale and P. Chapman. *Scenegraph Technologies: A Review*. University of Hull, 2007.

[2] S.S. Beauchemin and J.L. Barron. The Computation of Optical Flow. *ACM Computing Surveys*, 27(3):433–467, 1996.

[3] T. Capin, A. Haro, V. Setlur, and S. Wilkinson. Camera-based Virtual Environment Interaction on Mobile Devices. *Lecture Notes in Computer Science*, 4263:765–773, October 2006.

[4] S. Casas, P. Morillo, J. Gimeno, and M. Fernandez. SUED: An Extensible Framework for the Development of Low-cost DVE Systems. In *In Proceedings of the IEEE Virtual Reality 2009 (IEEE-VR'09). Workshop on Software Engineering and Architectures for Realtime Interactive Systems (SEARIS)*, pages 20–25. IEEE Computer Society Press, March 2009.

[5] M. Hakkarainen and C. Woodward. Symball: Camera Driven Table Tennis for Mobile Phones. In *In Proceedings of the 2005 ACM SIGCHI International Conference on advances in computer entertainment technology*, volume 265, pages 391–392. ACM Press, June 2005.

[6] J. Hannuksela, P. Sangi, and J. Heikkil. Vision-based motion estimation for interaction with mobile devices. *Computer Vision and Image Understanding*, 108:188–195, October 2007.

[7] A. Haro, K. Mori, T.Capin, and S. Wilkinson. Mobile Camera-based User Interaction. In *In Proceedings of the IEEE International Conference on Computer Vision (ICCV-05). Workshop on Human-Computer Interaction*, pages 79–89. IEEE Computer Society Press, October 2005.

[8] K. Hinckley, J. Pierce, M. Sinclair, and E. Horvitz. Sensing techniques for mobile interaction. In *In Proceedings of the 13th annual ACM Symposium on user interface software and technology sensing techniques for mobile interaction*, pages 91–100. ACM Press, November 2000.

[9] J. Hwang, G. J. Kim, and N. Kim. Camera based Relative Motion Tracking for Hand-held Virtual Reality. In *Proceedings of NICOGRAPH International 2006*. The Society for Art and Science, June 2006.

[10] M. Khri and D.J. Murphy. MARA: Sensor Based Augmented Reality System for Mobile Imaging Device. In *In Proceedings of Fifth IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 06)*. IEEE Computer Society Press, October 2006.

[11] HS. Liu and GKH. Pang. Accelerometer for mobile robot positioning. *IEEE Transactions on Industry Applications*, 37(3):812–819, May–June 2001.

[12] M. Mhring, C. Lessig, and O. Bimber. Video see-through AR on Consumer Cell-Phones. In *In Proceedings of Third IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 04)*, pages 252–253. IEEE Computer Society Press, November 2004.

[13] J. Rekimoto and Y. Ayatsuka. Cybercode: designing augmented reality environments with visual tags. In *n Proceedings of DARE 2000 on Designing Augmented Reality Environments*, pages 1–10. ACM Press, April 2000.

[14] M. Rohs and B. Gfeller. Using Camera-Equipped Mobile Phones for Interacting with Real-World Objects. *Advances in Pervasive Computing*, pages 265–271, 2004.

[15] SEMACODE. Choosing the best 2d barcode format for mobile apps. Technical report, SEMACODE Corporation, July 2006.

[16] L.G. Shapiro and G.C. Stockman. *Computer Vision*. Prentice Hall, 2001.

[17] D. Wagner, G. Reitmayr, A. Mulloni, T. Drummond, and D. Schmalstieg. Pose Tracking from Natural Features on Mobile Phones. In *In Proceedings of Seventh IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 08)*, pages 125–134. IEEE Computer Society Press, September 2008.

[18] D. Wagner, L. Tobias, and S. Dieter. Robust and Unobtrusive Marker Tracking on Mobile Phones. In *In Proceedings of Seventh IEEE and ACM International Symposium on Mixed and Augmented Reality (ISMAR 08)*, pages 121–124. IEEE Computer Society Press, September 2008.

[19] J. Wang and J. Canny. TinyMotion: Camera Phone Based Interaction Methods. In *In Proceedings of Conference on Human Factors in Computing Systems (CHI 06)*, pages 339–344. ACM Press, April 2006.

[20] J. Wang, S. Zhai, and J. Canny. Camera Phone Based Motion Sensing: Interaction Techniques. Applications and Performance Study. In *In Proceedings of the 19th annual ACM Ssymposium on User Interface Software and Technology*, pages 101–104. ACM Press, October 2006.

[21] D. Wigdor and R. Balakrishnan. Tilttext: Using tilt for text input to mobile phones. In *In Proceedings of the 16th annual ACM Ssymposium on User Interface Software and Technology*, pages 81–90. ACM

Press, October 2003.