Pintarolas, a Tangible Sketch Application

José Miguel Salles Dias
Miguel.Dias@iscte.pt

Juana Dehanov
juana@netcabo.pt

Carolina Cabral
carolinacabral@netcabo.pt

Antonio Carola
a.c@oninetspeed.pt

ADETTI/ISCTE, Associação para o Desenvolvimento das Telecomunicações e Técnicas de Informática, Edifício ISCTE, 1600-082 Lisboa, Portugal, www.adetti.iscte.pt

Abstract

This paper presents Pintarolas, a simple sketch application that uses tangible interfaces as Human-Computer Interaction modality. Totally sensor-less and cable-less interfaces (ordinary boardmarkers with fiducial markers attached), provide the means to support basic sketching tasks, such as drawing lines and sketching 2D primitive shapes (circle, triangle and square). The system requires an ordinary video camera linked to a PC and uses AR Toolkit for the handling of the Tangible Interfaces. Open GL is used for the graphical output and the CALI library, normally used for Calligraphic user interfaces development, is adopted in our case for 2D primitive shape recognition. A simple usability test was developed to assess the feasibility of this novel user interface in simple sketching tasks, showing that the users found the concept interesting and the tangible interfaces easy to operate.

Keywords

Tangible Interfaces, Calligraphic Interfaces, Multi-modal User Interfaces, AR Toolkit, CALI

1. INTRODUCTION

Sketch or Calligraphic-based Interfaces [Pereira 2003] are a now mature paradigm for Human-Computer Interaction, specially oriented to industrial design tasks. Such HCI modality tries to mimic the way in which users interact with paper documents and drawings. Calligraphic interfaces comprise a touch-sensitive screen or other surface where gestures 'drawn' with a special stylus, or a normal stylus whose position can be tracked, can be captured. Those gestures will then be interpreted as if they were being drawn in a sheet of paper. The popularity of this modality, is related to the nature of the task of writing/sketching on a flat surface, which is a skill shared by computer-users, or even unskilled users of all ages. This is due to the fact that the analogy with sketching in paper, allows for a more flexible and natural interaction than with other HCI modalities, such as the well known "mouse and keyboard" desktop metaphor. Commercial exploitation of the idea to recognize hand writing and sketching is illustrated, in products like Palm Personal Digital Assistants (Palm) or the Tablet PC.

With our work, our idea is to start from an available library that offers robust methods for the recognition of the scribbling of simple shapes, the CALI library [Fonseca 2000], but instead of using as physical interfaces, a stylus, linked by some sort of electronic media to the computer, we have adopt ordinary boardmarkers, that act as tangible interfaces. The pose (position and orientation) of these tangible interfaces need to be tracked, so that the sketching gestures can be capture and, for this purpose, we have used Computer Vision techniques. However, the solution to the general problem of tracking in 3D these physical objects, would require complex techniques of feature detection, feature tracking by template matching or motion flow and, possibly, the use of epipolar geometry in stereoscopy and 3D reconstruction [Trucco 1998].This approach although valid, would create undesired complexity in the found solution. Instead of solving the general problem, we have addressed in this paper, a simpler problem of detecting the 3D trajectory of fiducial marker centres attached to the boardmarkers and for that, we have used AR Toolkit [Kato 2001]. This can be seen as a transitional technology for this type of gestural-based HCI modality systems, since it simplifies the more complex and general problem of marker-less 3D tracking of physical objects such as boardmarkers.

With this work we are aiming at creating a tool that can be very useful to bring the sketching technique to a new dimension, the tangible sketching interaction. In synthesis, in this paper we present such tool, the *Pintarolas* system, which offers the Tangible Sketch HCI paradigm. The paper is organised as follows: in section 2, we provide a background in the libraries used: AR Toolkit and CALI. In section 3, we present our modular system functions and explain how the system operates. Section 4 explains how we track the hand sketching gestures, using the Tangible Interfaces and how the primitive shapes are recognised. Section 5 details a usability testing methodology and discusses the evaluation results obtained. Finally, in section 6, conclusions and future directions of research are given.

2. BACKGROUND

In the base of this system is the CALI library which is an open-source software library for the construction of Calligraphic Interfaces (intelligent interfaces based on sketches and gesture interaction). CALI is based on a simple Shape Recognizer that uses fuzzy logic, identifying Scribbles (multi-stroke geometric shapes) drawn with some type of user interface, such as a stylus. The Recognizer identifies elementary geometric shapes such as Triangles, Rectangles, Circles, Lines, and also some gesture commands, such as, Delete, Cross, Wavyline, Move, Copy and Tap. All the Shapes are identified independently of the changes in rotation, size or number of individual strokes.

For the computer vision layer and boardmarker tracking, our system adopts AR Toolkit, a C/Open GL-based open source library that uses accurate vision based tracking methods to determine the virtual camera pose information through the detection in real-time of fiducial markers and the calculation of reference frames associated to these markers.

3. SYSTEM ARCHITECTURE AND FUNCTION

Too deploy this new user interaction paradigm for basic sketching, a number of modules comprise our application, referred to as Pintarolas (see Figure 2). The **Computer Vision** Module includes a lower layer of **Live Video Input**, which feeds into **AR Toolkit**, which is able to recognize and track by computer vision means and in real time, the 3D pose (position and orientation) of fiducial markers centres, relatively to the virtual camera reference frame, whose mathematical model matches the real video camera capturing life images of the user issuing sketching gestures. In order to provide the required user interaction, dedicated system modules were developed, namely, **Tangible Interfaces** management and **Sketch Gesture Recognition**. The physical Tangible Interfaces are three ordinary pens with fiducial markers attached (see Figure 3): two for sketching purposes and a

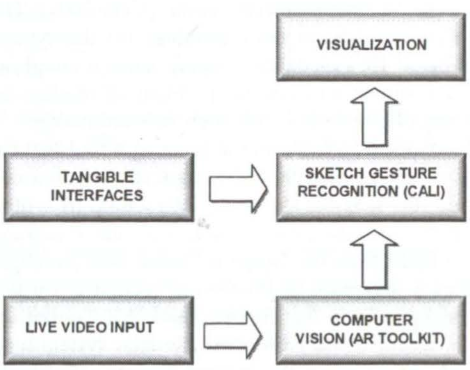

Figure 2 - Pintarolas System Architecture.

third for clearing the screen. With these interfaces the user is able to issue sketching gestures (drawing lines, circles, squares or triangles) without the need of being too precise. The **Sketch Gesture Recognition** module uses CALI, since this library offers robust methods that recognize the scribbling of simple shapes, such as lines, circles, triangles and squares, by interpreting the sketching gestures and improving the accuracy of the intended shape, as drawn by the user (see Figure 1).

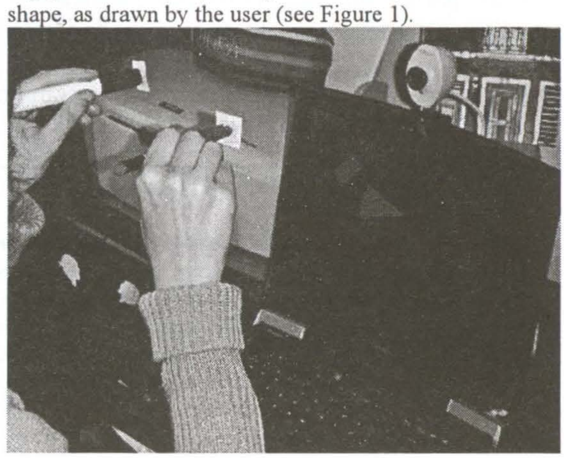

Figure 1 - User operating Pintarolas.

To operate with *Pintarolas* the user has three Tangible Interfaces available. Let us reefer to them as TI-1, TI-2 and TI-3. TI-1 is used as an indicator to where the user is pointing in the screen, in fixed or moving gestures. TI-2 is used, much like the left mouse button, when the user is ready to start a new drawing. The system assumes that the user is sketching with just one color, but more tangible interfaces could be added for sketching in different colors.

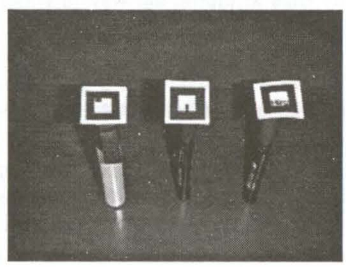

Figure 3 - Tangible Interfaces of Pintarolas.

To perform a drawing all it's needed to do is to show TI-1, locate on screen the place where the user wants to start the sketch and then show TI-2 to start sketching. When the sketch being drawn with the aid of TI-1 is finished, the user needs just to remove TI-2 from the camera line of sight. At this stage, the produced sketch is going to be recognized by the **Sketch Gesture Recognition** module and, if it matches one of the basic shapes early mentioned, the recognized primitive shape will be drawn, or else the drawn path will be kept (see Figure 4). Another interaction possibility is to delete the entire screen by showing a third Tangible Interface (TI-3).

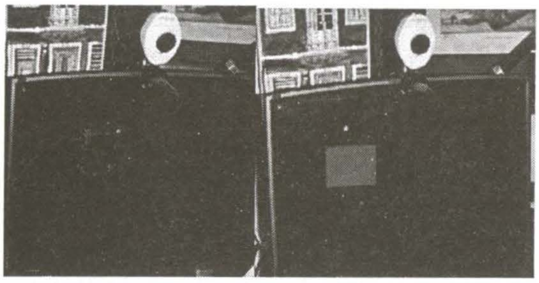

Figure 4 – Left: A square being drawn; Right: Recognition of the drawn square.

Each sketch made previously is kept on screen, unless it has been erased, by just pointing tangible interface TI-3 to any place on the screen.

4. TRACKING AND RECOGNIZING THE SKETCHES

4.1 Tracking the User Hand Gestures

The capture of the sketching gesture is made by detecting the trajectories of the centers of the fiducial markers attached to the Tangible Interfaces, which are provided by AR Toolkit. By projecting the positions of these centres in the 2D visualization window (by keeping the x and y coordinates of the centers, provided by AR Toolkit), we are able maintain the drawn path, which is basically a list of 2D points. Once we have these 2D coordinates, they are transformed to a given visualization window. In this window, each detected 2D coordinate will be painted as a small sphere, with a color that depends on the type of Tangible Interface being used (currently, only one color is supported).

4.2 Shape Recognizing

Once the user has removed the tangible interface TI-2, from the camera line of sight, we assume that the sketch has been made and the recognition processes can begin. From the moment TI-2 is shown until it is removed, all the 2D coordinates are kept in a structure called *stroke* (CALI library). We just use CALI sketches with one stroke, which corresponds to one scribble structure. A CALI library call is then invoked which returns either a recognized gesture (that has originated the sketching of a known primitive shape) or not. In the case of a positive answer, CALI will return the shape, which will be correctly drawn on screen using Open GL (Figure 4). The user can issue approximate gestures, that CALI can interpret and made more accurate. If the recognition is not made or the shape returned is Unknown, then the sketch is kept on the window as it was before the recognition process.

5. EVALUATION AND DISCUTION

We've conducted a series of usability tests to the system, to assess the feasibility of this novel user interface in simple sketching tasks and also to understand how common persons would interact with the system.

The usability testing experiment was designed to assess the usefulness of Tangible Interaction in a simple sketching application, by means of evaluating the way users perform simple sketching tasks on screen. These tasks consisted in the following basic operations:

1. Draw a line;
2. Draw a square;
3. Draw a triangle;
4. Draw a circle.

Each task was considered to be successful if the system was able to recognize the basic sketch (via the CALI library), within a maximum specified time (1 minute). The testing focussed on user performance evaluation and therefore, we have used two metrics. The first one was the time to complete the assigned task. This would give us a measure of how efficiently each of the individual basic sketches could be done via our interface, not only by itself but also in relation to the other basic sketching tasks. To identify problems with the interface, the other metric, was the number and type of errors made by each individual subject. The types of error we have considered beforehand were:

1. False positives;
2. Failing to complete the task;
3. Taking too long to complete the task (more than 1 minute);
4. Other errors (to be seen during the tests).

The experiment was run on a sample of 10 unpaid users in one run. The users were students in their early 20's, selected randomly from undergraduate courses at ISCTE, Instituto Superior de Ciências do Trabalho e da Empresa, in Lisbon. Of the 10 persons, 5 were males and 5 were females. All the users that have interacted with the system had never been in contact with Augmented Reality or Sketch-based interaction systems. The facilitator gave initially a brief explanation of how to operate the system and showed the four tasks; drawing a line, drawing a square, drawing a triangle and drawing a circle. Subsequently, one at a time, the test subjects were evaluated. For each subject, a total of 2 minutes were given for him/her to properly train each of the tasks and their various aspects with the help of the facilitator. The user was then asked if he/she understood what was meant to do. Then, after concluding training and making sure the subject was ready, he/she was given 1 minute (maximum) to complete each of the proposed tasks.

5.1 Time Analysis

In Figure 5 we depict the observed task times. It's clear that the "Draw a Circle" task, was the one that took longer to complete, and with a figure that seams to be too much for the nature of the task. It's worth noticing also that the CALI library as some specific shape requirements, to consider a given sketch as of a circular shape.

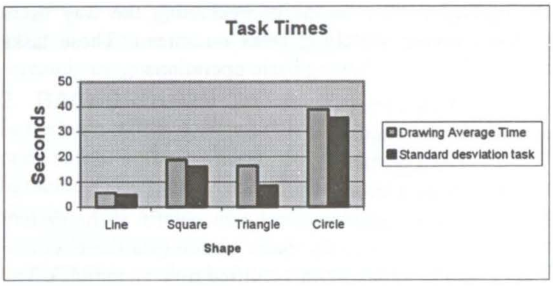

Figure 5 -Task Times

As it can be seen on Figure 5 the time needed do draw a line is quite reduced, because it's a very simple procedure to accomplish. As for the squares and triangles, the time needed to perform the drawing is quite alike, this happens because the effort made to draw them only varies due to the difference of number of sides and also, because the CALI library can easily deduced these types of shapes.

5.2 Task Analysis

Analyzing the observed errors per task, we can see that the most difficult sketch to draw and be recognized was the one of circular shape, which matches the finds in the Time Analysis. Almost all of the false positives where observed when the users have tried to make small shapes.

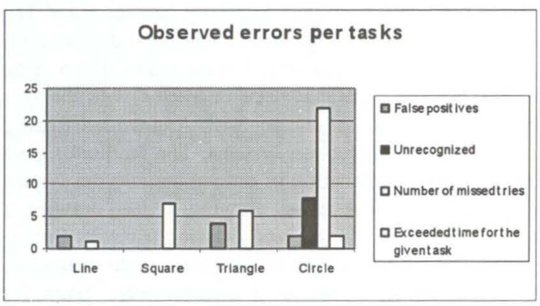

Figure 6 - Observed errors per tasks.

We've also observed that all the users made their best effort to complete successfully the tasks proposed. As a consequence the majority of them adapted really quickly to handle correctly the two Tangible Interfaces. As it can be seen on the Figure 6 although some users found it hard to complete with success the given task, they always found a way to do it, since only two people weren't able to perform a circle, being the only shape that has exceeded the maximum given time. The number of missed tries can be explained due to the lack of dexterity of the users, and the fact that some times they forgot that the recognition was only successful when performing the sketch in only one stroke. The main comments made by the users were the problems associated with the marker tracking related to the lighting conditions, since some times the system wouldn't identify the Tangible Interfaces poses. In general all the users thought that it was a simple system to use, and that it demonstrated an interesting concept.

6. CONCLUSIONS AND FUTURE DIRECTIONS

In this paper, we have described a simple sketch application that uses tangible interfaces as Human-Computer Interaction modality. By analysing the evaluation results of simple usability tests performed for this application, which conclude that users easily adapt to this type of HCI, we have shown that simple to use tangible interfaces, are appropriate for basic sketching tasks. However, some analysis is still missing and is planned for future activities, such as the precision of the sketched drawings, and the observed rate of errors. The authors are also aware that the use of two tangible interfaces for the sketching task, is a diversion from the natural way of sketching. So, as a future line of research, we are considering the integration of other modalities (like voice interaction), with the current one, towards the deployment of a more usable and natural interface, Regarding other future directions for the evolution of this system, we can think of improving the marker tracking technique, by including in-house adaptive threshold algorithms that perform the image binarisation, prior to the marker tracking proper. The main improvements however should be done in gradually introducing the functionalities that the real paint applications do have. In this line of activity, we could increase the number of shapes that are recognised by the system and add the recognition of letters. Although currently in a very early stage of development, we can think of potential applications of *Pintarolas*, for example, in areas like teaching and children gaming. In teaching scenarios using large displays, this type of interface could better the blackboard metaphor, by introducing a new way of interaction between the teacher and the students and by increasing also the mobility of the teacher. In the children games area, this basic concept could be use to create a series of learning games for small children, for them to improve and practice their calligraphy and to increase their sense of interactivity and their interest in course subjects. The system could also be used for people with motor disabilities, with the purpose of trying to improve their impairments.

7. ACKNOWLEDGEMENTS

The authors would like to thank professors Joaquim A. Jorge and Manuel da Fonseca from IST, for the kind provision of the CALI library and Pedro Santos, from ADETTI, for his help with AR Toolkit.

8. REFERENCES

[Fonseca 2000] Fonseca, M., J., Jorge, J., "*$C_{AL}I$: A Software Library for Calligraphic Interfaces*", 9º EPCG, 2000.

[Kato 2001] Kato, H., "Developing Applications with ARToolkit", SIGGRAPH 2001 Course Notes 27, 2001.

[Pereira 2003] Pereira, J. P., Jorge, J., Branco, V. A., Ferreira, F. N., "Calligraphic Interfaces: Mixed Metaphors for Design", Tenth Workshop on the Design, Specification and Verification of Interactive Systems, DSV-IS 2003, Funchal, Portugal, 4-6 June 2003.

[Trucco 1998] Trucco, E. Verri, A., *Introductory Techniques for 3-D Computer vision*, Prentice Hall, ISBN: 0-13-261108-2, USA, 1998.