D-TOUCH: A CONSUMER-GRADE TANGIBLE INTERFACE MODULE AND MUSICAL APPLICATIONS

E. Costanza
Media Engineering
Department of Electronics
University of York
York
UK
e.costanza@ieee.org

S. B. Shelley
Media Engineering
Department of Electronics
University of York
York
UK
sbs102@york.ac.uk

J. A. Robinson
Media Engineering
Department of Electronics
University of York
York
UK
jar11@ohm.york.ac.uk

ABSTRACT

We define a class of tangible media applications that can be implemented on consumer-grade personal computers. These applications interpret user manipulation of physical objects in a restricted space and produce unlocalized outputs. We propose a generic approach to the implementation of such interfaces using flexible fiducial markers, which identify objects to a robust and fast video-processing algorithm, so they can be recognized and tracked in real time. We describe an implementation of the technology, then report two new, flexible music performance applications that demonstrate and validate it.

Keywords

Tangible Media, Physical Objects Interface, Video Analysis, Music User Interface.

1. INTRODUCTION

The power of physical or "tangible" interfaces has been particularly well demonstrated in Video-Augmented Environments (VAEs) ([14], [13], [12], [9]). But VAEs require expensive equipment such as data projectors and specially designed interaction objects. In low-budget environments like schools and homes, interfaces must be implemented with conventionally-equipped personal computers and everyday objects. We are interested in extending the paradigm of tangible interfaces to educational and recreational applications, and so seek ways of realising them on consumer-grade equipment.

We confine our attention to tangible media applications that do not generate localized outputs – or, at least, for which the localization of the outputs at the interaction objects is not required – such as those that output only audio. The physical configuration of objects provides all the visual feedback needed.

The main requirement for implementing tangible media input is robust and fast analysis of the movement of physical objects. Our tangible media environment consists of a small area, usually on a table top, viewed by a web cam, near which are placed the computer speakers. It is assumed that the computer screen will not be used in tangible media applications. For interaction we expect different applications to use different simple moveable objects. We therefore need a way for the system to recognize these. Our solution is to use a library of generic fiducial symbols that the system has high probability of recognizing and tracking accurately and quickly even in adverse illumination and with partial occlusion.

2. CONTEXT AND PRIOR ART

The use of physical objects as tangible interactors offers several benefits noted by previous workers. These include: enhanced multiuser interaction ([5], [14], [3]; enhanced spatial awareness through 3D vision and kinaesthetic memory and improved use of spatial reasoning skills ([8], [14]).

A fiducial-based approach to both wearable and projected augmented reality has been advocated by Rekimoto and others ([10], [7], [13], [11]) with barcode, matrix code and character recognition methods demonstrated. However, whereas these methods are based on geometrical feature extraction, generally followed by template matching, our approach relies on the topological structure of the markers. This provides a number of advantages discussed in section 4, and prompts careful co-design of the fiducial and the image processing method.

We use topological image processing to achieve real-time fiducial identification and localization. Inspired by a region adjacency graph approach by Clarke and Johnston [6], we have developed a novel region adjacency tree algorithm. This is a simplification of previous topological approaches to recognition, allowing us to use simple adaptive thresholding of input video images and fast tree traversal. The result is a method that is both accurate and robust.

3. MAKING OBJECTS INTERACTIVE

To support a wide range of tangible user interfaces we need to keep track, in real-time, of all the interactive tokens present within the interface area in terms of their identity and position. This information will then be mapped to a system reaction or state. As both the object positions and identities characterise the interface state, several objects of different types can be present in the interface at the same time.

Figure 1. *Objects labelled with fiducial symbols***.**

In our system, each interactive object is marked with a two-level fiducial pattern, as illustrated in Figure 1. These fiducial symbols are then detected by means of the video processing algorithm outlined below. In general, one application can include several types (or "classes") of interactive objects. To each of these classes corresponds a different fiducial symbol, which is stuck or painted on the physical objects to be used. All the symbols defined in the same application constitute what we call the "palette" (or "family") of fiducial markers of that application. In a first stage of processing, the positions (x,y coordinates) of all the fiducial symbols that are present in the interface are determined. Then, if the palette includes more than one class of objects, a second step separates the results of the first stage, identifying all the members of each class.

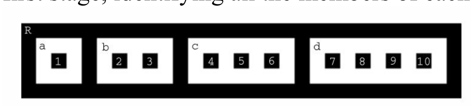


Figure 2. *A fiducial marker.*

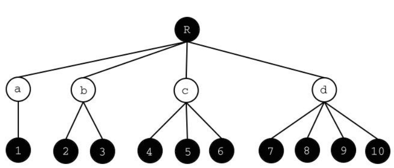


Figure 3. *Region Adjacency Tree for the marker in Figure 3.*

The first stage of fiducial recognition is based on the topological property of region adjacency rather than on shape (in contrast to existing systems [11], [7], [13]). In any application, all the elements of a palette have the same topological structure. This is stored in the form of a bipartite tree, as shown in figures 2 and 3. Each image acquired from the camera is reduced to binary using a local adaptive threshold selection, and analysed in terms of the adjacency of its connected components. This information is subsequently compared with the structure of adjacency of the fiducial symbols, looking for matches. Details of the algorithm for constructing the image region adjacency tree and matching against the fiducial tree are given in [1].

The second stage of fiducial processing (distinguishing between different classes) uses object geometry in a limited way. By leaving one of the branches fixed as a reference/pivot, the disposition of the other branches can be varied; this generates an alphabet of different symbols that are equivalent from a topological/region adjacency point of view. An example is shown in figure 4.

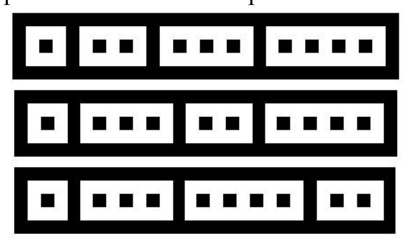


Figure 4. *Different fiducial markers with the same topological structure.*

The geometry of the fiducial symbols is otherwise free, as is their colour. Both may be used to give meaningful cues, or to meet other design objectives. To achieve good performance, the structure of the fiducial symbols has to be simple enough for the search to be fast but complex enough not to be confused with natural objects that may be in the field of view of the camera. We conducted experiments to measure the performance of different fiducial structures and find the simplest topology that would provide good protection against both false positives (i.e. the number of regions erroneously classified as a fiducial symbols) and false negatives (i.e. the number of fiducial symbols not recognized).

A full discussion of the experiments can be found in [1]. Overall resilience to false positives was very high even with heavily cluttered backgrounds. Some of the fiducial structures were never falsely detected in the entire set of 10000 images. Balanced fiducials (i.e. Those that have a constant number of leaves in each branch) gave better performance than unbalanced ones with the same number of leaves.

The geometry of the fiducial markers can be designed for compactness, additional tolerance to deformation and partial occlusion, protection against misclassification or aesthetic considerations related to the application context ([11]). In general these constraints will require a trade off. Tuning of these parameters is done by considering the particular needs of each application. For instance if the interaction is to take place in a controlled restricted area, with no external elements, such as on a structured board, the requirements regarding false positives can be lowered to the advantage of other factors (e.g. compactness). On the other hand if the interface has to be integrated on an existing working table full of other ordinary objects, additional protection against false positives could be crucial to the effectiveness of the interface. Also the shape and size of the physical objects used in each application will require careful consideration; this can be the starting point for the design of the fiducial symbols as they must fit the surface available on the objects.

The geometric design of fiducials is therefore a complicated question. In practice, we have successfully used square, circular, rectangular and sketchily drawn fiducials. For many applications, including those we discuss later, the end user could simply print out or draw stick-on fiducial markers and attach them to any objects of his/her choice.

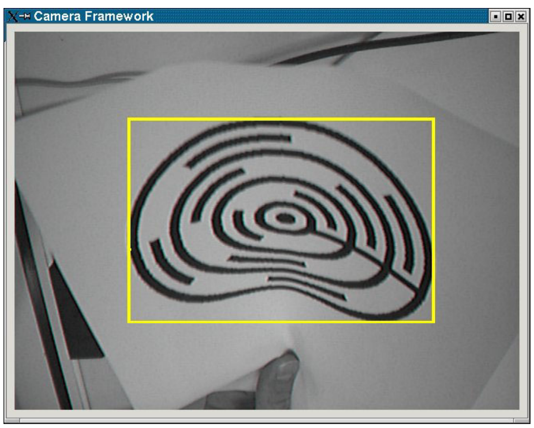


Figure 2. *Different fiducial markers with the same topological structure.*

4. IMPLEMENTATION: *D-TOUCH*

We have implemented the methods described in section 4 in *d-touch*, a software module that can be used to experiment, rapidly prototype and develop a wide variety of tangible media applications. Figure 5 shows detection of a large fiducial in *d-touch*; later figures show small fiducials. As well as working over a wide range of scales, *d-touch* provides good tolerance to luminance change, shadowing and distortion, including warping distortion, as in figure 5, so that fiducials can be bendable.

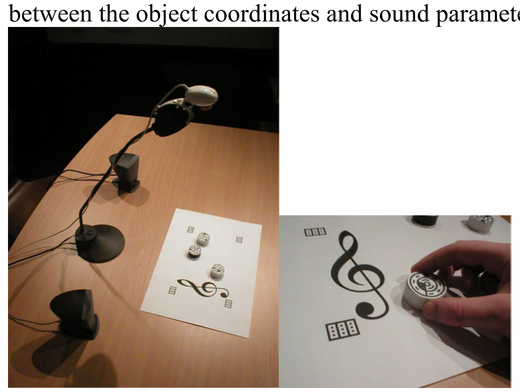
d-touch allows interaction to take place on any surface, using the coordinate system of the camera frame, or some other reference can be employed. This can be easily implemented by attaching fiducial symbols on a number of fixed points in the interactive area. This arrangement allows the system to triangulate the camera position with respect to the augmented surface. The position of the camera can then be used as control variable.

The recognition of the topological structure of the symbols is used as a filter to select quickly the areas of interest of the picture, then other simple image processing steps are employed to detect additional information. The first step can also provide by-product information that can be directly reused for further processing.

The spatial resolution that can be obtained even with consumer grade products is high. As an example, it is possible to use a palette of 6 objects as small as 26x26mm in an area of about 300x400mm (about the size of an A3 sheet). The test was carried out using a Philips PCVC740K "ToUCam Pro" web cam, which has a CCD sensor with resolution 640x480, and is in the region of £50 to the end user. As a reference for the evaluation of these results, the "ARToolkit" [7] requires square markers of size about 88x88mm to operate from the same camera distance.

5. MUSIC APPLICATIONS AND TANGIBLE MEDIA

Two example musical applications have been implemented using d-touch: an "augmented musical stave" and a "tangible drum machine". Both use simple mapping between the object coordinates and sound parameters.



Figures 3 and 7. *The Aumgented Stave*

In the first application (Figures 6 and 7) physical representations of musical notes (cylindrical objects about 4 centimetres in diameter and 1.5 cm in height) can be placed on a stave drawn on a sheet of paper to compose simple melodies or to teach the score notation to children. As soon as a note is placed on the stave, the corresponding sound is played by the computer. Different kinds of objects represent different note lengths (semi-quavers, quavers, crotchets, semi-breves, etc..).

Similarly, in the "tangible drum machine" (Figure 8) toy-like square based blocks are used to represent drum sounds and can be arranged on a grid printed on a piece of paper (displayed to provide a visual cue for the user). This yields an interface where the user can build complex drum rhythms and naturally adjust them by moving the physical blocks. The interactive objects position is mapped to the time sounds are played within the loop and the type of drum sound, while different interactive object classes correspond to different sound volumes (to allow accents in the beat).

Both applications are described and compared with other electronic musical instrument in [2].

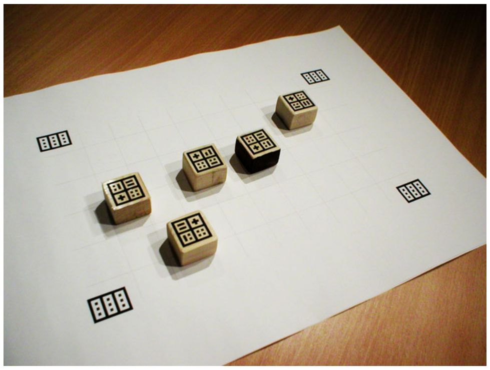


Figure 8. *The Tangible Drum Machine.*

6. CONCLUSIONS AND FUTURE DIRECTIONS

We have introduced *d-touch*, a module that can be used to develop a range of tangible interface applications, requiring only consumer-grade hardware. The pattern recognition algorithm on which it is based has been outlined.

The use of video processing as opposed to RF tagging, proposed in other systems, makes the interaction objects very economical, versatile and easily replicable. Interactive objects can be made as safe as it is required by the application environment, by selecting appropriate sizes and materials. Because fiducial symbols are recognizable when flexed or curved they can be attached to soft objects, such as foam toys that can be used by preschool children.

At the same time it is possible to use our interface for "hybrid" applications, where the arrangement of the real objects in physical space is used to control and modify, in real-time, the position of virtual objects displayed on a computer monitor. In this configuration the tangible interface would be confined to act more as an input device (rather than a full interface). This kind of approach has been shown for example in [4]. Even though such applications step back to the use of a computer monitor to mediate the interaction, they keep some of the advantages of pure tangible interfaces.

In the future, we plan to develop and test other applications based on the *d-touch* interface. In particular we aim to explore the potential for live performance of music and the chance to embed the interface into more complex augmented reality applications. Moreover we are interested in applications targeted at users with special needs and for education.

A formal series of subjective tests for the existing musical applications is under consideration to compare them to equivalent programs that use graphical user interfaces.

7. REFERENCES

[1] Costanza, E., Robinson, J., A Region Adjacency Tree Approach to the Detection and Design of Fiducials, accepted for publication from *Vision, Video and Graphics, 2003, Bath, UK, July 2003.*

[2] Costanza, E., Shelley, S., Robinson, J., *Introducing Audio d-touch: a Novel Tangible User Interface for Music Composition and Performance, submitted to the 6th international conference on digital audio effects (DAFx-03) London September 2003.*

[3] *Fitzmaurice, G., Ishii, H., Buxton, W. Bricks: Laying the Foundations for Graspable User Interfaces. Proceedings of the Conference on Human Factors in Computer Systems (CHI'95). p442-449. 1995.*

[4] *Goslin J,spaceCube,http://*www.jeremygoslin.com

[5] *Ishii, H., and Ullmer B., Tangible bits: towards seamless interfaces between people, bits and atoms. In CHI'97, pages 234 -- 241, 1997.*

[6] *Jonston D., Clarke A., A Vision Based Location System Using Fiducials,* accepted for publication from *Vision, Video and Graphics, 2003, Bath, UK, July 2003.*

[7] *Kato H. and Billinghurst M., Marker Tracking and HMD Calibration for a Video-based Augmented Reality Conferencing System, Proc. of the IEEE & ACM IWAR '99, pp.85-94, 1999.*

[8] *Patten J., and Ishii H., A Comparison of Spatial Organization Strategies in Graphical and Tangible User Interfaces, in Proceedings of Designing Augmented Reality Environments (DARE '00) , (Elsinore, Denmark, April 12-14, 2000), pp. 41-50 .*

[9] *Pinhanez C., Augmented Reality with Projected Interactive Displays. Proc. of International Symposium on Virtual and Augmented Architecture (VAA'01), Dublin, Ireland, 2001.*

[10] *Rekimoto, J., Navicam: A Magnifying Glass Approach to Augmented Reality, Presence, Vol 6, No 4, August 1997, pp 399-412.*

[11] *Rekimoto J. and Ayatsuka Y., CyberCode: Designing Augmented Reality Environments with Visual Tags, Proceedings of DARE 2000, 2000.*

[12] *Robinson J., Robertson, C., The LivePaper system: augmenting paper on an enhanced tabletop, Computer & Graphics,* ***25****: 731-743, 2001.*

[13] *Underkoffler, J. and Ishii, H. Illuminating Light: An Optical Design Tool with a Luminous-Tangible Interface. Proceedings of CHI '98, April 1998.*

[14] Wellner, P., Interacting with paper on the DigitalDesk. Communications of the ACM, **36(7)**:86--96, July 1993. 28.