Multitouch Sensing for Collaborative Interactive Walls

Alessandro Soro[1]**, Gavino Paddeu**[2]**, and Mirko Luca Lobina**[3]

[1] CRS4, Italy, asoro@crs4.it
[2] CRS4, Italy, gavino@crs4.it
[3] CRS4, Italy, mlobina@crs4.it

Abstract: In this paper we explain the design of t-Frame, a hardware/software architecture that allows the implementation of multiuser interactive wall. t-Frame brings multi-touch sensing to a generic display by means of low cost digital video cameras. The design of t-Frame is illustrated in detail, together with a prototype installation. We show how t-Frame differs from other approaches and discuss our findings, together with a plan of future research and improvements.

1. Introduction and Motivation

Interactive walls are a special kind of computer applications that deliver a highly impressive, shared view of information, and are suited to many exciting applications, ranging from workgroup collaboration to pervasive computing and entertainment. t-Frame is a low-cost hardware/software architecture that enables multi-touch interaction on a generic display. Specifically, t-Frame is intended to be used in large size, multi-user interactive walls. With respect to other approaches t-Frame can be installed as a pointing device on any flat surface regardless of size, shape and display technology.

t-Frame allows researchers in the field of human-computer interaction to set up with minimal effort an environment for experimenting in the field of computer supported collaborative work and tangible user interfaces: the goals of the project can be summarized as follows:

1. provide multi-user and multi-touch interaction to any display, regardless of the specific technology of the display itself;
2. minimize both the cost of installation and maintenance, using standard hardware and simplify the installation and calibration procedures;

3. be applicable to very wide installations: modern hardware allows the creation of interactive walls several meters long using cluster systems or multi-head graphic adapters.

By contrast, the most adopted technique to implement a multitouch displays is, at the moment, based on FTIR (Han, 2005), and requires the adoption of an expensive high resolution IR camera, ambient IR screening and is in practice bounded to rear projected screens.

t-Frame requires less space than other technologies, can be easily transported because it does not require a single-piece touch surface (the prototype implementation described in this paper consists of a display tile, although any other display solution can be exploited) and, most important, can adapt to the size of the display seamlessly, without sensibly affecting the performances.

2. Related Work

Pioneer work on multi-touch sensing devices can be tracked back to the mid eighties, see for example (Lee, 1985)(Krueger, 1985)(Kasday, 1984). An overview of the evolution of multi-touch technologies is maintained in (Buxton, 2008). Given that multi-user interaction is a straightforward extension of multi-touch sensing, the obvious playground in this field consists in displays capable of accommodating a number of users, such as tabletop and wall-size displays. (Dietz, 2002) and (Wilson, 2005) are examples of the former, (Wilson, 2004) and (Dempski, 2005) of the latter. Several techniques have been exploited to implement multi-touch sensing devices: (Dietz, 2002) consists of an array of antennas whose signals get transmitted, through the body of the user, to a receiver that elaborates touch events. Among optical techniques (Wilson, 2004) exploits stereo cameras to compute hands position, but the cameras are located behind the semi-transparent screen, thus the system is bounded to front/rear projected display. The same holds in (Han, 2005), which relies on an infrared camera that captures the light that escapes the display surface when finger contact occurs. In (Oka, 2002) the optical sensor is located above the display surface, and thus the hands of the user(s) stay between the camera and the screen. Although feasible for the interactive desk described, this approach would be not practical for interactive walls, since the body of the user would in general cover the hands from the viewpoint of the camera. The system described in (Denlinger, 1988) exploits an approach similar to t-Frame: the cameras are arranged around the screen and the position of the fingers is determined through triangulation, but the cameras are located on the corners of the screen: in t-Frame the particular arrangement of the cameras limit the complexity of the algorithms involved in finger triangulation and allows the system to scale in size almost indefinitely.

3. t-frame Design

A t-Frame installation consists of a set of cameras arranged on the plane of the display. In a typical installation of an interactive wall, cameras are aligned on top of the screen, facing downwards, as shown in Figure 1a. However, cameras are not bounded to a fixed position or orientation, and can be arranged anyhow, as long as they lie on the same plane of the screen and the exact position and orientation of each camera is known with respect to screen coordinates. In order to simplify the setup of the installation t-Frame provides a calibration facility that computes the exact position of each camera, this operation involves three steps: i) every camera takes a snapshot of its field of view, no touch must occur during this process, and saves it as a known background; ii) the user is requested to point an horizon in the background image of each camera: the horizon must be specified as close as possible to the surface of the screen; iii) the user is requested to point with her finger three given points on the screen: the position of the cameras is triangulated exploiting the images captured at each touch. Once the installation has been correctly calibrated each camera enters a continuous loop in which still images are captured at the maximum possible frame rate. Periodically the images are compared to gather touch events that are then pushed in an event queue that applications can poll and consume. In the following the two most critical steps of t-frame are presented.

3.1 Finger Triangulation

The frames captured by each camera are elaborated to spot touch events. A frame is compared against the known background along the line of the horizon: when a significant difference is found, the algorithm assumes that the background is covered by a finger touching the screen (see Figure 1b) and measures its position. The position of the finger is reported as the angle under which the finger is seen with respect to the center of the field of view. To do this the exact aperture of the field of view must be known, since we only can take measures in terms of relative positions, i.e. counting pixels in the image. Figure 1c shows how the exact position of a finger is computed in screen coordinates: when a finger touches the surface of the screen it is seen by every camera whose field of view covers that position. Additionally every camera sees the finger under a given angle. Then, computing the position of the finger is as easy as calculating the intersection of two lines passing through the position of any given camera (that is known from calibration), and intersects the axes of the screen under the given angle, which is a matter of bare trigonometry. With a single touch and only two cameras this approach doesn't differ from stereo vision, but by exploiting several cameras t-Frame can easily recognize an arbitrary number of touches.
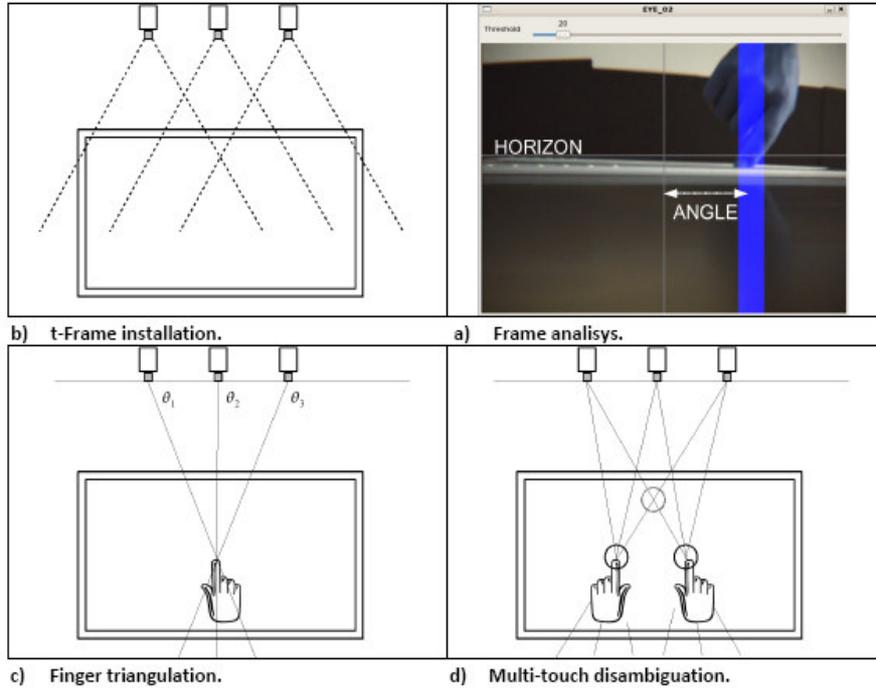
Figure 1 t-frame design.

3.2 Multi-touch disambiguation

Consider the situation schematized in Figure 1d. The user touches the screen with two fingers: each camera sees its background covered by two distinct obstacles and computes two angles. We can therefore trace two lines for each camera; the intersections of such lines determine the exact position of the fingers, but we need a strategy to exclude false spots. To this end the algorithm of multi-touch disambiguation compares every intersection (that is a candidate finger) against any camera that has that specific point within its field of view, and checks if the candidate finger is compatible with the image seen. In the case of Figure 1d the candidate finger highlighted with a grey circle is the result of the analysis of the frames taken by the leftmost and rightmost cameras. In order for this to be accepted as a finger-touch, it must be confirmed by the frame taken by the central camera. The comparison against the central camera shows that this is a fake finger, as expected, and that only the two spots highlighted in black circles represent true finger-touches. The number of cameras needed to disambiguate a given configuration depends on many factors. In general with N cameras we can always disambiguate N – 1 finger-touches, but in practice fewer cameras are enough to monitor many finger-touches. In fact, in many cases, false candidates

fall (largely) outside of the display area and can be excluded. Additionally finger-touches tend to follow predictable paths, and heuristic techniques can be exploited to distinguish false candidates. It can be shown that the complexity of the algorithms is linear with respect to the number of cameras installed, and depends primarily on the number of fingers that are seen at once by a given set of adjacent cameras. A formal proof of this result is anyway outside the scope of this paper.

4. Prototype Implementation

The t-Frame design and algorithms have been tested in a prototype installation, consisting of a 60" wide display tile, driven by a high-end graphical workstation. The optical apparatus consisted of 9 VGA cameras (25 frames/sec). The system performs very well in terms of efficiency, since the algorithms involved in the image analysis don't comport a significant overload. In addition, the exact computation of fingers' position only involves some basic operations. Our experiments show that multi-camera triangulation based on background subtraction is a suitable technique for implementing a multi-user & multi-touch interactive wall. The event models that are common in computer systems, based on clicks, drags and so on, have been designed for single pointer operations. Although such a model can be extended to support multi pointer interaction (such as in MPX http://wearables.unisa.edu.au/mpx/) this cannot be other than a first step towards a new, commonly agreed, event model specifically designed for multi-touch widgets.

5. Conclusions and Further Work

We have shown the design and a prototype implementation of t-Frame, a computer architecture that deploys multi-touch sensing capabilities to any display surface. This novel framework presents several encouraging aspects: i) it is a completely new approach for implementing a multi-touch strategy; ii) it is fully scalable from small touchable surfaces to large interactive walls; iii) it can be applied also to no-display surfaces, as geo-charts or posters; iv) it is basically a solution of compact dimensions, easily portable and based on cheap hardware. The next step of the t-Frame project is the clarification of several shortcomings we have observed during experiments. First of all the precision of the pointing system strongly depends on the resolution of the installed cameras. At the same time also a certain temporal data-ambiguity is observable due to the frame rate of the cameras. Finally, a new paradigm should be introduced for the event management. In fact, the typical model of events introduced with the mouse is not suitable for a multi-touch framework. Apart from the clarification of these problems, we are particularly interested in experimenting in the near future with no-flat screens, such as panoramic, or even cylindrical, displays, either front and back projected.

Our intention is to deploy t-Frame as collaboration platform for complex information retrieval and manipulation, but besides our initial scenarios, many other applications are emerging that could take advantage from the technology described in this paper, in particular, outdoor installation of public interactive walls, to use as high-tech exhibition spaces, and entertainment platforms.

References

Angioni, M., Demontis, R., Deriu, M., De Vita, E., Lai, C., Marcialis, I., Pintus, A., Piras, A., Soro, A., Tuveri F.: User Oriented Information Retrieval in a Collaborative and Context Aware Search Engine. WSEAS Transactions on Computer Research, ISSN: 1991-8755, vol. 2(1), pp. 79-86, (2007).

Buxton , W.: Multi-Touch Systems that I Have Known and Loved. http://www.billbuxton.com/multitouchOverview.html. Accessed 15 Apr 2008.

Denlinger, B. M. Ambient-light Responsive Touch Screen Data Input Method and System. U.S. Patent 4782328. (1988)

Dempski, K., Harvey, B.: Supporting Collaborative Touch interaction with High Resolution Wall Displays. In 2nd Workshop on Multi-User and Ubiquitous User Interfaces. Jan. 2005 at IUI'05. (2005)

Dietz, P. H., Darren, L. L.: Multi-User Touch Surface. U.S. Patent 6498590. (2002)

Han, J. Y.: Low-cost multi-touch sensing through frustrated total internal reflection. In Proceedings of the 18th Annual ACM Symposium on User interface Software and Technology (Seattle, WA, USA, October 23 - 26, 2005). UIST '05. ACM Press, New York, NY, 115-118. (2005)

Kasday, L.: Touch Position Sensitive Surface. U.S. Patent 4484179. (1984)

Krueger, M. W., Gionfriddo, T., and Hinrichsen, K.: VIDEOPLACE—an artificial reality. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (San Francisco, California, United States). CHI '85. ACM Press, New York, NY, 35-40. (1985)

Lee, S., Buxton, W. & Smith, K.C.: A Multi-Touch Three Dimensional Touch-Sensitive Tablet. In Proceedings of the 1985 Conference on Human Factors in Computer Systems, CHI '85, San Francisco, April, 1985, 21-26 (1985)

Oka, K., Sato, Y., Koike, H.: Real-Time Tracking of Multiple Fingertips and Gesture Recognition for Augmented Desk Interface Systems. FGR 2002: 429-434. (2002)

Wilson, A. D.: TouchLight: an imaging touch screen and display for gesture-based interaction. In Proceedings of the 6th international Conference on Multimodal interfaces (State College, PA, USA, October 13 - 15, 2004). ICMI '04. ACM Press, New York, NY, 69-76. (2004)

Wilson, A. D.: PlayAnywhere: a compact interactive tabletop projection-vision system. In Proceedings of the 18th Annual ACM Symposium on User interface Software and Technology (Seattle, WA, USA, October 23 - 26, 2005). UIST '05. ACM Press, New York, NY, 83-92. (2005).