Mobile Hand Tracking Using FPGAs for Low Powered Augmented Reality

Wayne Piekarski[1], Ross Smith[1], Grant Wigley[2], Bruce Thomas[1], David Kearney[2]
Wearable Computer Laboratory[1] / Reconfigurable Computing Laboratory[2]
School of Computer and Information Science
University of South Australia
Mawson Lakes, SA, 5095, Australia
{wayne,wigley,thomas,kearney}@cs.unisa.edu.au, ross@tinmith.net

Abstract

Many augmented reality systems use general purpose computing hardware to perform tasks such as rendering computer graphics, video overlay, and vision tracking. This can result in systems being large and bulky due to the hardware complexity required and the power consumed. We have developed a hand tracking solution in a reconfigurable computer, which reduces power consumption and transfers some of the processing into specialised hardware. This paper presents a summary of the design and its implementation details.

1 Introduction

Most augmented reality (AR) research that has been published to date relies on the use of general purpose computing hardware. This hardware is typically larger in size and consumes more power than dedicated hardware which is designed specifically for a task. When building wearable computers, portability and power consumption are important issues that need to be addressed. In our Tinmith-Metro modelling system [3], a backpack computer is worn by the user (shown in Figure 1), and tracked gloves are used to perform pointing and command selection. We previously developed a tracking system which uses Firewire cameras and software [4], but this was very processor intensive to operate.

In this paper, we describe our system which is based on new dedicated hardware components, and is shown in Figure 1 and Figure 2. A Celoxica RC200 reconfigurable computer [2] was used, which contains a Xilinx Virtex II 1000 field programmable gate array (FPGA), random-access memory, an LCD panel for debugging, video capture chips, and an RS-232 port. Rather than using the FPGA for all image processing, the RC200 contains dedicated circuits for commonly performed tasks. Video frames are captured from a head worn PAL resolution camera and the coordinates of markers attached to the user's thumbs are calculated. The same video camera signal is also sent to a GrandTec MagicView video overlay device to provide video see-through AR. The advantage of this configuration is that there is no need for the laptop to process the video stream. The measured power consumption of the RC200 without the LCD display is 4 W, while the video overlay device consumes 1.9 W. We foresee a possible future AR implementation not requiring a laptop, instead built using only specialised hardware for the task.

2 Background

An FPGA consists of an array of uncommitted logic that can be configured by the end user through a form of hardware programming. Programming these devices can be performed using languages such as VHDL or Verilog, but this can be difficult and time consuming, especially for inexperienced users. We have used the Handel-C [1] programming language, which uses a C-like syntax to express circuit designs. Although this C-like syntax is easier to understand than VHDL, algorithms must still be redesigned for the parallel nature of the architecture to achieve the best performance. The use of high-level languages such as Handel-C is still an open research topic

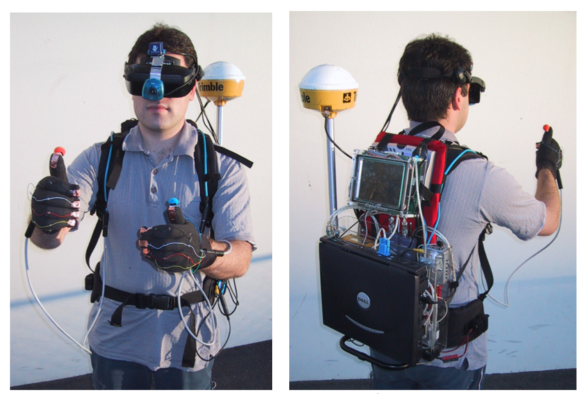

Figure 1 - Tinmith backpack with the RC200 and video overlay devices mounted at the top, and the user wearing gloves with coloured markers attached for the hardware vision tracker

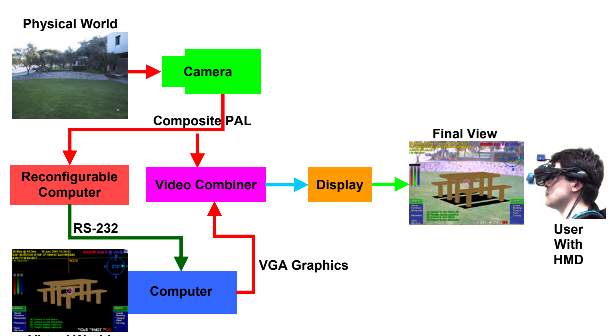

Figure 2 - Overall system schematic, showing the RC200 integrated with the general purpose laptop, and video AR implemented using hardware overlay of the two video signals

however, and is much less mature than comparable general purpose languages such as C and Java. FPGA programming requires a detailed understanding of the circuit architecture and timing, but allows the development of hard real-time systems. Algorithms may be broken up into parallelised components, but strict synchronisation is required to guarantee un-corrupted results.

We studied a number of algorithms to determine which would be most suitable for implementation in hardware. The main strength of an FPGA is that it can exploit algorithmic parallelism with little change in the amount of power required. Algorithms were selected based on their ability to be implemented completely within the FPGA, without needing to access slower external memory.

3 Implementation

To separate the coloured ball from the scene, we search for the ball rather than trying to search for the background which is unpredictable and always changing in outdoor environments. Rather than thresholding the colour of the ball in RGB colour space, we use YUV data from the onboard SAA113H capture chip so that brightness information is contained in its own a separate channel. The thresholding algorithm searches for a range of colours in UV space, and uses a wide tolerance on the Y channel so it will perform well in a wide range of lighting conditions.

After thresholding has separated out candidate pixels, we apply a centre of mass algorithm onto the image to find the centre of these pixels in the scene. Any large blobs (such as the ones worn by the user) will be clearly visible against any background noise. By performing multiple threshold and centre of mass algorithms in parallel, it is possible to search for multiple coloured blobs simultaneously. The centre of mass algorithm is particularly well suited to FPGA implementation because regions of the image can be processed in a series of parallel pipelines with the results combined later.

After the blob centres are found, the RS-232 serial port is used to transmit the XY coordinates to the host machine for rendering to the AR display using software such as Tinmith-Metro. The RC200 captures frames at PAL 50 Hz refresh rates, and is processed by the FPGA at 25 Hz in real-time due to the interlacing of the video signals. Since the custom hardware has no other overheads, the delay is the refresh rate plus the RS-232 transmission delay.

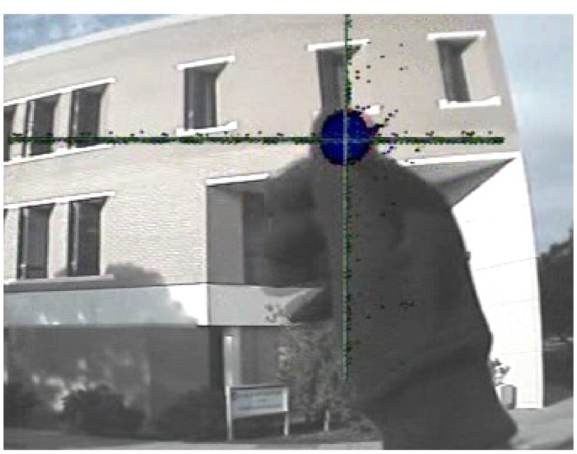

Figure 3 - Debugging output from the RC200 showing image thresholding and the calculated marker centre point, combined with the camera view of the outdoor environment

4 Results

We have tested our system outdoors and the results are much more robust than our previous fiducial marker based system [4]. Since the tracker does not record any state information from frame to frame, it recovers from tracking failures instantly and does not need to reacquire the markers. We performed a number of experiments to find colours that are not commonly present in typical outdoor environments to lower our rate of false detections. We found that using highly saturated colours (in particular, red and orange) produced reliable blob thresholding. By using marker balls with a furry surface, we were able to remove specular highlights which would result in the failure of the tracking algorithm. Compared to our previous tracker, our new implementation is able to more easily survive extreme lighting conditions such as when the sun is almost in the field of view of the camera, which is quite common. The main weakness we have noticed is during twilight conditions, when the camera is unable to distinguish colours in the environment as easily. Figure 3 shows the debugging output from the RC200, indicating the detected pixels as the coloured blob, with a cursor indicating the detected centre. The tracker always produces results that are within the bounds of the marker, unless there is a greater amount of noise from the environment. The serial data stream is used to supply a cursor for our existing Tinmith-Metro modelling system [3], which operates exactly the same as previously presented.

5 Conclusion

This paper presented our initial research into a hand tracking system for wearable computers implemented using an FPGA and custom hardware. We have demonstrated the feasibility of implementing a suitable algorithm and tested it in real outdoor conditions. In the future, we hope to further improve miniaturisation and power consumption by offloading more tasks to dedicated hardware.

6 References

[1] Celoxica. *Handel-C.* http://www.celoxica.com

[2] Celoxica. *RC200 Development Board.* http://www.celoxica.com

[3] Piekarski, W. and Thomas, B. H. Interactive Augmented Reality Techniques for Construction at a Distance of 3D Geometry. In *7th Int'l Workshop on Immersive Projection Technology / 9th Eurographics Workshop on Virtual Environments,* Zurich, Switzerland, May 2003.

[4] Piekarski, W., Avery, B., Thomas, B. H., and Malbezin, P. Integrated Head and Hand Tracking for Indoor and Outdoor Augmented Reality. In *IEEE Virtual Reality Conference,* Chicago, Il, Mar 2004.