Pre-commercial 3D Digital TV studio

J.-C. Nebel and W. P. Cockshott V. Yarmolenko, E. Borland

1 Introduction

In March 2002, the authors and Svenja Hoff submitted a paper titled an ‘Experimental 3D Digital TV studio’, which was published in this journal in February 2003 [1]. Recently, we had the honour to be awarded by the council of the IEE the A H Reeve Premium for that paper. Therefore, it fell natural to offer the journal this short communication paper describing the latest technical developments regarding our 3D Digital TV studio and its commercial applications.

2 Technical improvements

Better image quality

The acquisition of powerful white strobe lights (UNILUX HiLighter, 4000Lux) allowed the optical capture set-up to be radically improved. Instead of using overhead projectors, the optical quality of which is poor, equipped with white strobes to provide the texture pattern and conventional studio lighting for the colour images, we adopted an alternative scenario. The random dot pattern is now projected using 35mm slide projectors equipped with good quality lenses. The new strobe lights are used, in synchronisation with the colour cameras, to drown out the texture pattern for the colour images. Moreover, they operate at a frequency of 50 Hz to reduce significantly the discomfort of the subject facing them.

Grid parallelisation

Ideally, data captured by the system should be processed at 25Hz; however, the operations are very CPU-intensive. Since the processes are inherently easy to parallelise, the main challenge is to the sheer volume of data being generated and moved. Our goal was to use Grid technology to acquire the resources needed. However, the dynamic nature of the process creation, dispatching requirements and the necessity of supporting a reconfigurable inter-process communications network required an extension of existing Globus tools. Therefore, we had to implement a new Java interface. That software, called Jp [2], was modelled on the primitives of the p-calculus.

Parallelisation experiments were conducted using the process architecture shown on Figure 1 [3]. Each capture client runs a number of job management threads and, as soon as a new pair of monochrome images is read, control of processing is passed to one of these. The newly active thread immediately attempts to acquire an available matcher server process. Once a connection is accepted, input data (~600Kbytes) is streamed from the client. When the matching process has completed, the disparity map (~3.6 Mbytes) is returned to the client, and the matcher waits for a new connection. This arrangement ensures that all the server hosts are equally utilised and each runs no more that one server task at a time.

The server hosts fell into 2 categories: local machines attached to a 100Mbps Ethernet LAN and a remote IBM 16 CPU host linked via a 1Gbps pipe. The network is a critical factor and ultimately limits the amount of parallelism possible: calculations suggest that for our configuration, parallelisation of about 150 should be achievable. With the CPU resources available (up to 43), a speedup of an order of magnitude was demonstrated for a similar increase in the number of processors. With gigabit interfaces a further order of magnitude increase is theoretically achievable. Overall, parallelisation of the order of 10^3 would seem feasible with a sufficiently high bandwidth network. Such a level of CPU power, while currently not available, would make real time processing of 3D images a realistic possible.

3 Applications

3D motion analysis

Generally, the animation of 3D characters is based on the animation of hierarchic rigid bodies defined by a skeleton which is the supporting structures for a polygonal mesh that represent the character skin. The smooth deformation of the skin around articulations is achieved by displacing vertices according to the motion of the different bones of the neighbourhood. The process of associating vertices with weighted bones is called skinning. It is a manual and tedious procedure based on successive tries until convergence towards an artistically pleasing result. We decided to tackle that process by developing a semi-automatic method based on the analysis of sequences of 3D data generated by our system.

Since the 3D models have different topologies, the generation of range flows is a prerequisite to a high density analysis of non-rigid motions. The matching algorithm used to generate range maps was applied to track features over time to produce optical flows. By combining series of range maps with their corresponding optical flows, we could generate the required range flows, see Figure 2. From that data, bone rotations could be extracted and weight distributions of skin vertices were generated [4].

A plug-in was written for one of the main commercial animation packages, 3D Studio MAX^{TM}. Our skinning software was then integrated within the workflow of the commercially led project, VMAN, which aims at offering an intuitive authoring system allowing the creation, animation, control and interaction of 3D virtual characters [5].

Facial animation

Through the PGPGrid project, a joint venture between the University of Glasgow, Peppers Ghost Productions and the Edinburgh Parallel Computing Centre, we aim to demonstrate the use of dynamic 3D data to drive CG animation [3]. In addition to providing a more productive workflow, animation generated in this way should prove to be much more believable than animation generated by traditional means. In particular we emphasise the subtle and complex facial movements that we can capture which are often lacking in artificially generated animations.

The most important step is moving from a set of discreet models to a single unified model which an animator can interact with consistently across frames. This is achieved by mapping a generic model to the captured data using a process called conformation [6]. This relies on finding corresponding positions of features on both 3D models. We have opted to use a set of the points derived from the MPEG-4 standard for facial animation. These landmarks are positioned manually for the first frame of each sequence. Then a simple prediction of where the landmarks will be in each subsequent frame is made, and positions are adjusted by the user. We are currently working on the automation of the process using tracking techniques similar to the one presented in the previous section. Finally, noise present in the scanned data is reduced by applying a high-frequency filter on the motion of vertices of the unified model. Moreover, data quality is also improved by considering spatial information from neighbouring frames, see Figure 3 [7].

Sequences of unified models can then be imported into animation packages such as 3D Studio MAX^{TM}. They can also be used to generate MPEG-4 Facial Animation Parameters (FAP), which allow the transposition of the animation of one face to another.

4 Conclusion

Our 3D Digital TV studio is now a reliable system providing high quality data which can be handled by the Grid. It has attracted much interest from the animation industry. Several applications are at the pre-commercialisation stage since collaborations with the animation industry have shown that they would benefit from our system by integrating it in their work flow to produce efficiently subtle and realistic animations.

References

[1] Cockshott, W.P., Hoff, S., Nebel, J.-C.: ‘Experimental 3D Digital TV studio’. *IEE Proceedings - Vision, Image & Signal Processing*, 150(1):28-33, February 2003.

[2] Yarmolenko,V., Cockshott, P., Borland, E., Graham, P., Mackenzie, L.: ‘Jp interface: A Java implementation of the pi-calculus for GRID computing’, *Proceedings to the International Middleware Conference (MGC)*, Toronto, Oct. 2004.

[3] Cockshott, P., Mackenzie, L.M., Yarmolenko,V.: ‘PGPGrid Project’. *Global GRID Forum*, Berlin, 2004.

[4] Nebel, J-C., Sibiryakov, A.: ‘Range flow from stereo-temporal matching: application to skinning’. *Proceedings of International Conference on Visualization, Imaging, and Image Processing*, 2002.

[5] Sibiryakov A., Ju X., Nebel J.-C.: ‘A new automated workflow for 3D character creation based on 3D scanned data’. *LNCS 2897, Virtual Storytelling*, 155-162, 2003.

[6] Ju, X., Siebert, J.P.: ‘Conformation from generic animatable models to 3D scanned data’, *Proceedings of Numérisation 3D/Scanning 2001*, Paris, France, 2001.

[7] MacVicar, D.W., Ford, S., Borland, E., Rixon, R.C., Patterson, J.W., Cockshott, W.P.: ‘3D Performance Capture for Facial Animation’. *Proceedings of second Int. Symposium on 3D Data Processing, Visualization, and Transmission*, 2004.

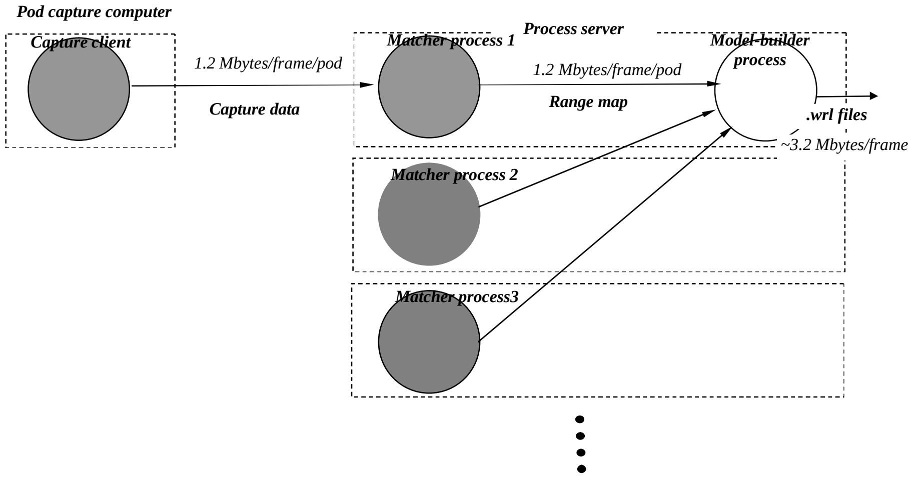

Figure 1: Basic Parallelisation data flows for process server

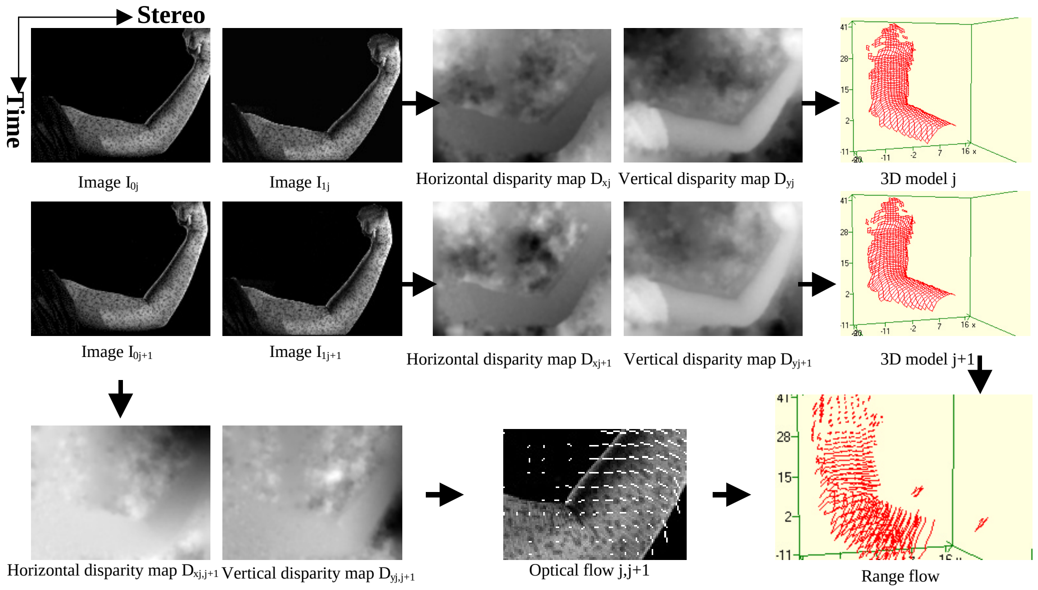

Figure 2: Range flow generation

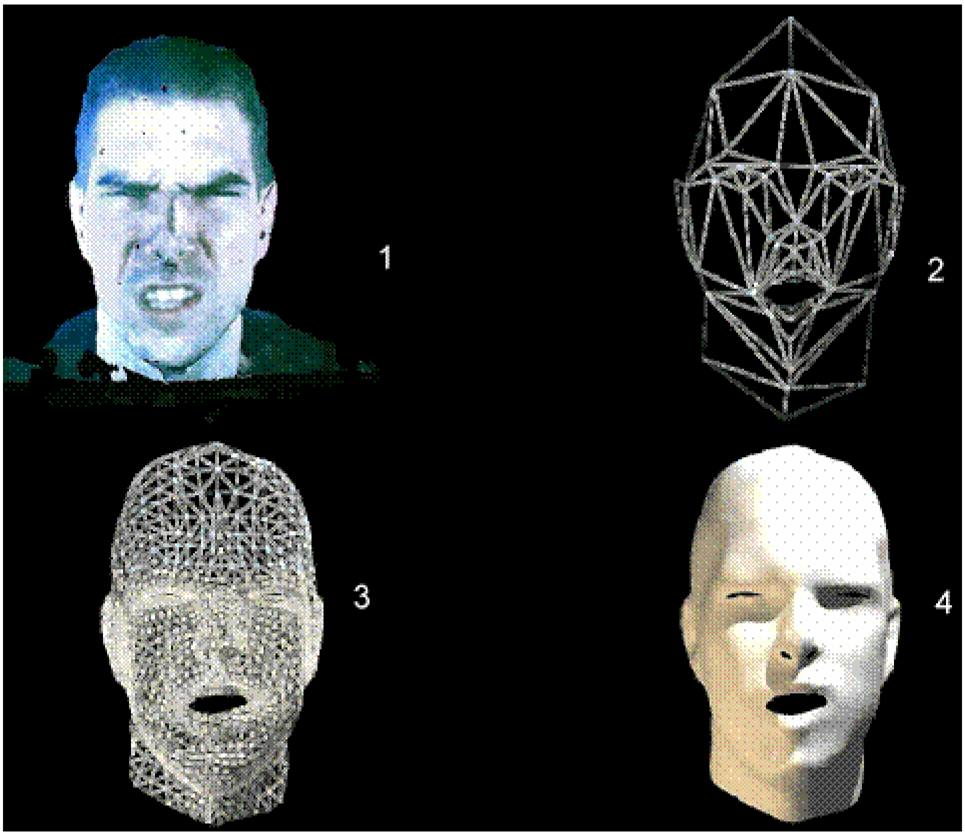

Figure 3: Process steps: 1. raw non-anatomical 3D model, 2. Fitted landmark mesh, 3. fitted complete facial mesh, 4. Facial mesh rendered