Technologies for characters in computer games

Alan Watt and Steve Maddock
Department of Computer Science, University of Sheffield, UK
a.watt@dcs.shef.ac.uk, s.maddock@dcs.shef.ac.uk

Abstract

Much of the output of computer games development falls into a set of tried and tired genres. It is rare to see a game breaking out of this spiral of stagnation. We suggest that improving the quality of character animation may be the answer, and that by creating believable characters, new gaming experiences may appear. In the paper, we discuss some of the issues surrounding the creation of believable characters and present some of the work done at Sheffield in this direction.

1 Introduction

With the revenue of a successful game now exceeding that of a Hollywood film it seems pertinent to examine the state of games and try and extrapolate towards the future. There are other parallels with Hollywood than just the cash and the main one is the persistence of genres, particularly the first person shooter. At the moment we seem to be locked into a cycle of hardware advances being driven by games industry demand, resulting in sequels of existing games that are more complex geometrically and are better lit, but otherwise much the same.

What the hardware advances have not done is to engender a vibrant culture of genre innovation. From one point of view the games culture is stagnant and reflects the apparent love for synthetic violence beloved by Hollywood films. In this paper we examine the reasons for this, from a technology point of view, and point out the areas which could enable us to break out of this cycle.

We begin by proposing that the current limitations of character animation are surely one of the main reasons for this stagnation. We propose that human beings are interested in interacting with virtual characters, human-like or otherwise (and the success of the Sims tends to bear this out); but the characters must be believable and this is the main issue.

1.1 Believability and the *uncanny valley*

Believability in human-like characters is a difficult issue and research done decades ago is often quoted when this issue is debated. This was carried out by the Japanese roboticist Masahiro Mori [Mori70], who coined the phrase *uncanny valley* to describe the negative-going excursion, indicating a repulsive response, on the graph shown in Figure 1. The graph purports to show that we can only increase the believability of robots, and by extension, we assume, game characters, up to a certain point. We seem to be able to increase the human-like appearance of a synthetic character up to a threshold beyond which the empathy decreases sharply and the character becomes strongly repulsive. As the character becomes more human-like the empathy returns.

Perhaps the fear of falling into the uncanny valley is the reason why companies such as Pixar and Disney have concentrated on cartoon-like character stylisation. The message from games is that we are very much on the left of the valley and moving to the right is currently not viable. We still cannot produce characters that are indistinguishable from real humans, despite having crossed this believability barrier in rendering – we can render scenes consisting of objects that an observer would find difficult to differentiate from reality. However, certain fragments of character behaviour are believable, the obvious example being captured motion (MoCap).

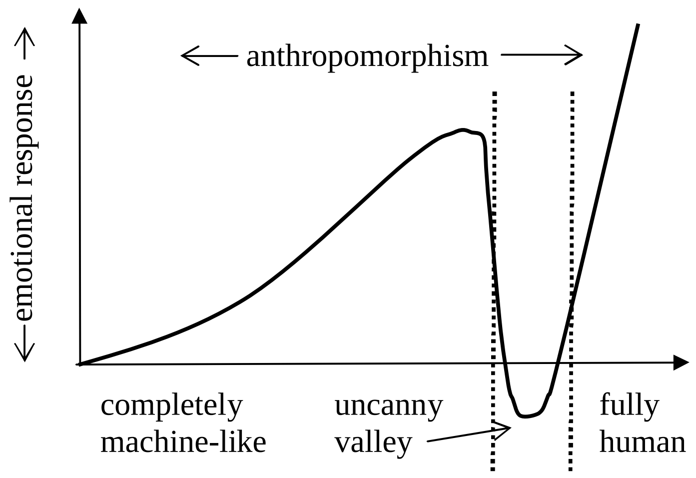

Figure 1: Emotional response against robot realism.

1.2 Believability in Games

It is difficult to define what exactly we mean by believability in games. Perhaps an 'industry' definition would be a game that that exhibited sufficient attraction that it became a success in the market. Leaving aside that contentious question we can say that the three predominant factors that determine the believability of a computer game can be loosely categorised as:

- **Rendering quality**: all the factors that contribute to the overall look of the game including the art creation.
- **Motion quality:** the quality of the animation of whatever type.
- **Artificial Intelligence:** a very loose categorisation meaning the overall game concept, interaction with player and whatever other factors that do not fall into the first two categories.

In what follows, we examine the first two factors only, in relation to character animation.

2 Rendering Quality

In many ways rendering is a solved problem. The increasing march of hardware means that we can have astonishingly complex levels, dynamically lit and inhabited by detailed characters. However, as far as human-like characters are concerned, the conventional rendering model is to author a skin mesh which is then wrapped onto bone matrices. This approach produces unsatisfactory deformations on the skin of the character, be it clothing, or flesh, and its current inertia is undoubtedly influenced by the overwhelming success of MoCap, which by definition needs to drive a bone or skeleton model. Thus, in a sense, the motion quality achieved by MoCap has stifled increased rendering quality in human-like characters.

However, recently reported research already includes real-time solutions for important rendering improvements such as more realistic reflection models for skin (involving sub-surface scattering), the animation of cloth and the animation and rendering of hair.

Realistic skin rendering in the games industry is currently achieved by artists using carefully created texture maps or photo textures. This works well, but significant improvements can be gained by using more realistic reflection models that attend to sub-surface scattering – one of the physical interactions that distinguishes light-skin interaction from light-plastic interaction. Research into such models has a long history in off-line rendering (see, for example, [Hanr93]). Recent real-time approaches tend to be imitative rather than physically based (in the same way that the Phong reflection model is imitative rather than being physically based). A good example of this type of approach is [Gree04].

Cloth animation has followed the same kind of path, beginning with complex off-line models that used, for example, catenary curves to simulate draped fibres [Watt92] and progressing to real-time mass-spring models [Land99] and simulations based on particle systems [Oshi01].

Animating and rendering hair is difficult because we need high complexity (consider the number of hairs required) convincing animation and an anisotropic reflection model. A popular much used reflection model is the Banks model [Bank04], which can be used to render common 'strand' objects like hair and grass. Hair rendering is perhaps the most researched area of the three and many real-time algorithms are reported (see for example [Gibb01] and [Mert04]).

3 MoCap for the motion of human figures

Again the games industry view this as a solved problem and this is attested to by not only its visual quality but also its advantages from the workflow point of view over authored animation.

Although the first uses of MoCap were to be found in the TV and Film industry, the games industry was the first to embrace the technology as a routine method for animating human-like characters. Now the games industry accounts for much of the total usage and almost all games that employ such characters use MoCap to drive the animation.

The quality and convenience of MoCap is convincing. Consider Figure 2. Although the figure is a simple stick skeleton the resulting animation from MoCap data exhibits complex and expressive motion. This 'ghost' sequence shows sets of superimposed frames at equal sampling intervals. Providing the motion is sampled at a sufficiently high rate, the technology captures all the subtlety of the human performance. Examine the illustration; here is easily seen (leg to the right) the deceleration in the motion. Using key frame animation from two keys positioned at the start and end of the ghost frames would not result in the correct motion. To imitate such a sequence using a key frame system, the animator must position the keys appropriately so that the required accelerations and decelerations are present in the final sequence. It is precisely the difficulty and cost of accomplishing this that motivates the use of MoCap data.

One of the disadvantages of MoCap is that it requires post processing. This is extremely important and demands much effort. The operations are mainly 'clean-ups'

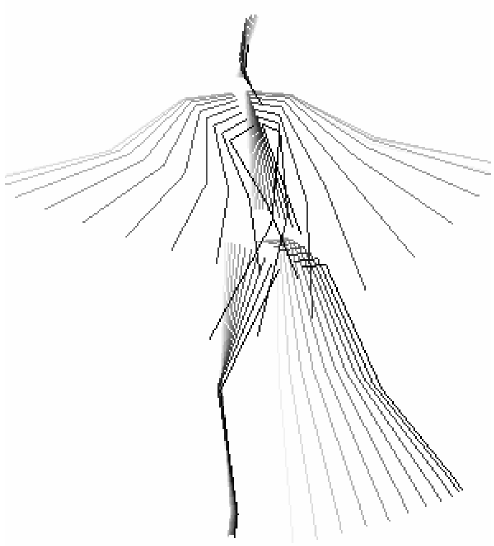

Figure 2: MoCap data applied to a stick-figure skeleton

and involve noise removal, filling in gaps caused by a marker going out of view for a period, and eliminating confusion when two markers become coincident in a view. Another important low-level processing operation is converting the data – which specifies the position of markers as a function of time – into the joint rotation form suitable for driving a skeleton hierarchy.

Time-consuming and complex operations, such as retargeting are at the moment, performed off-line. This situation is likely to change as the technology develops with more processing moving into the real-time domain. For example, in a report published in 2001, Shin et al [Shin01] handle both the low-level pre-processing operations and re-targeting operation in real-time. Their application is computer puppetry – by definition a real-time application. Here the movements of a performer are mapped to an animated character and a prototype system has been used successfully to create a virtual character for a children's TV program, as well as a virtual news reporter. Computer puppetry may find applications in the future in multi-player applications; but for now the lesson that all conventional MoCap processing can be done in real time has significance for the games industry.

The two main drawbacks of MoCap technology are:

- We can only activate pre-recorded scripts in a game - an obvious statement, but one that needs emphasis. Although a large number of sequences are stored and selected in real-time according to game logic, this is still an inherently limiting process. We would like to have facilities so that MoCap sequences continually adapt themselves to the developing game, altering themselves and producing new sequences from existing material.
- MoCap data is only 'valid' for a virtual character who possesses the same scale as the real human from which the data was recorded. When we try to use the data on a character of different scale, we encounter problems. This is known as the retargeting problem.

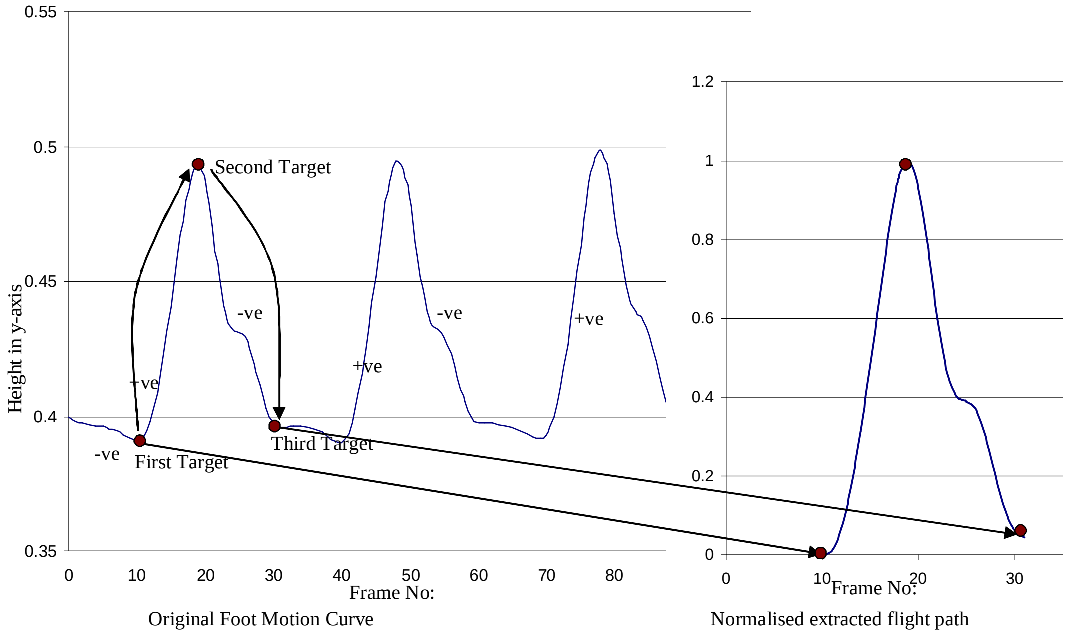

Figure 3: Gradient-based extraction of foot flight from MoCap data

There are many other subtle problems in the technology. A quality consideration, which has meant less take up from the film industry (c.f. the games industry), derives from the fact that sampling the motion of points on the surface of the skin of a character does not lead to a completely accurate script with which to animate a skeleton composed of rigid links. Currently the aesthetic demands in character animation are lower for games than for film and this has meant the wholesale take-up of MoCap technology in games.

We conclude that MoCap will persist as a successful technology because of the quality it achieves, its inertia as an established industry and also because it is a powerful marketing tool. You can MoCap famous footballers, for example, and exploit this in your marketing campaign.

The industry has been reluctant to adopt alternatives to MoCap and resolutely sticks to either MoCap or authoring of curves (represented as functions) to drive cycles of particular parts of the body. Whichever data is used to drive the motion, it needs to be adapted to retarget the data to a different-sized figure or to alter the motion to fit the environment (so that, for example, feet do not slide across the floor), or to offer individualisation of motion (so that, for example, a footballer's behaviour could gradually react to an injury). This can be done using inverse kinematics (IK), which is now possible in real time. Meredith and Maddock [Mere04] use a weighted IK approach to alter MoCap data to introduce a leg injury to a character. Figure 3 shows the flight curve of a foot extracted from a MoCap sequence using gradient analysis. The extracted data can then be adapted in real-time to drive new walks on a figure. Figure 4 shows the parts of the walk cycle as it is used on a figure. Figure 5 illustrates the final motion on a skeleton.

Whilst such techniques can be used in real-time it is probable that they would only be used for important, foreground characters, with less important, background characters having less processing power spent on them.

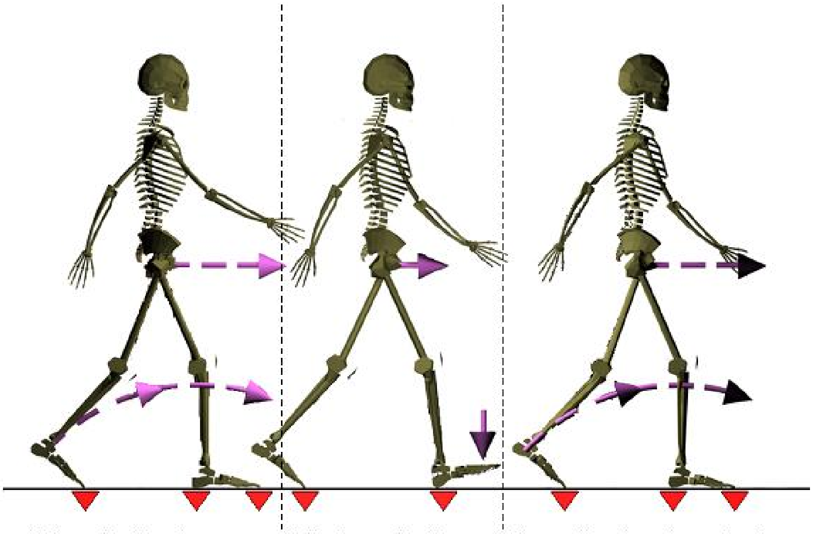

Figure 4: Parts of the walk cycle. Each frame represents the start of the cycle with the arrows pointing in the direction of travel the node will take until it reaches the start of the next part of the cycle. The red triangles represent plants of the character's limbs.

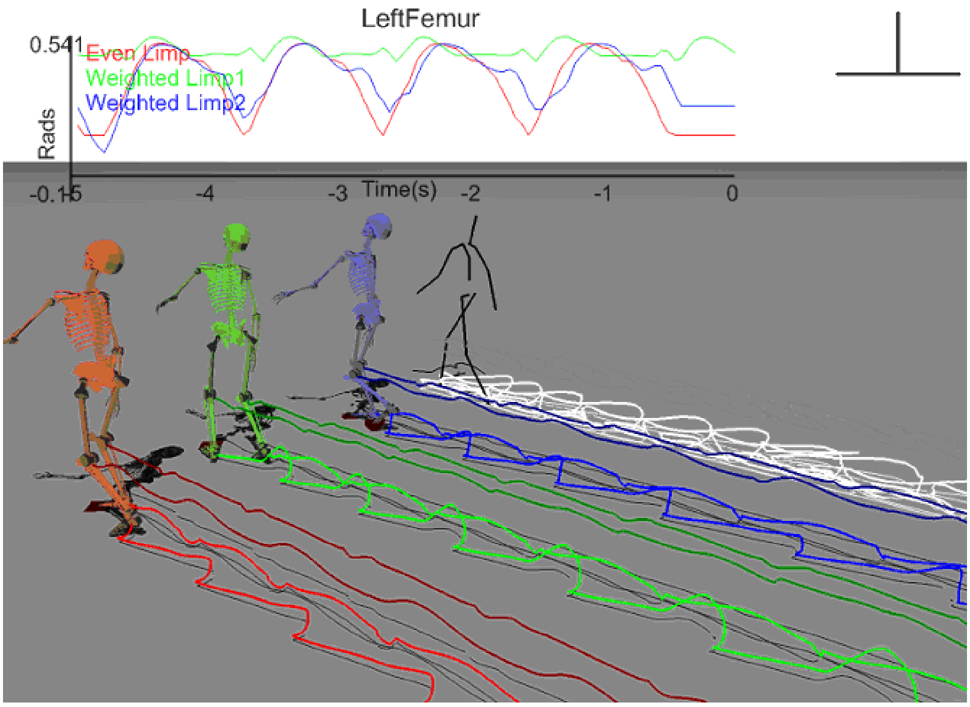

Figure 5: Injury simulation – limping left leg

4 Facial animation

If believable character animation is the last frontier in computer games, within that field facial animation has proven to be the most challenging. The difficulties are obvious and do not need to be stated in much detail. Facial expressions are quite subtle and this implies that we must use a very high resolution mesh. And thus controlling the motion of, say, 10^4 mesh points using a number of control points two to three orders of magnitude less is a difficult problem and one that is still a vigorous research area.

- *Integration*: expressive visual speech involves both lip motion and facial expressions that occur simultaneously across the entire face. A believable model has to integrate them into a whole driven by appropriate parameters.
- *Identity/Uniqueness*: a second difficulty arises from the uniqueness of faces. Ideally we would like an underlying deformation model which admitted different 'masks'. Currently the most common manifestation of this approach is (photographic) texture mapping onto a generic polygon mesh. The mesh vertices are then animated. The identity problem is absent from, or does not concern us to the same extent, in other aspects of human animation (body and fabric for instance).
- *Rendering Quality*: the detail of expressions (wrinkles etc.) is difficult to render convincingly using standard geometric and shading models. At a finer level of detail rendering of skin texture is also not currently accomplished to high quality. In the absence of convincing photorealistic quality we have sometimes resorted to cartoon-like characters.

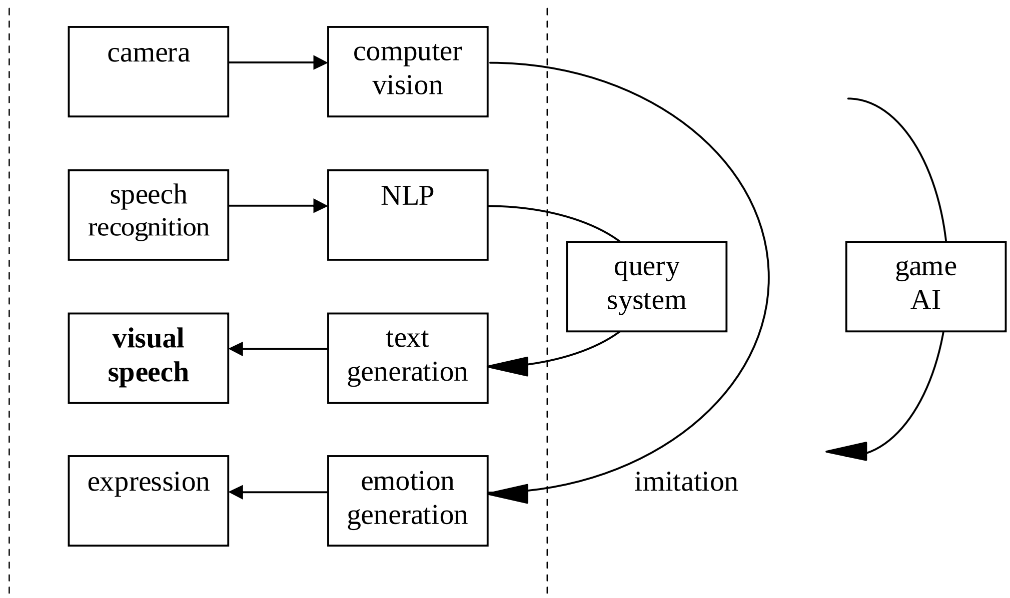

Figure 6: Parts of an anthropomorphic interface.

Perhaps the most important aspect of facial animation is visual speech and it is the case that this area of high importance across all computer applications has lagged behind progress in audio speech synthesis. Figure 6 emphasises this point and shows a visual speech module as part of an 'anthropomorphic' interface. Such interfaces must eventually predominate in man machine communication; the implications of this technology for computer games are obvious.

Using MoCap for facial motion is difficult because, unlike the motion of limbs, the motion of the face surface is subtle and involves important small-scale movement, such as creases and wrinkling, which is impossible to capture from the limited spatial density of markers. This, of course, contrasts with the quality of captured body motion whose success has spawned an entire industry.

On the other hand we can use MoCap to record large-scale motion of the lips and jaw. Because of these difficulties the current methodology in games is for artists to model static poses as keys or morph targets – see, for example, [Bees04]. These then form keys and inbetween expression motion is interpolated in real-time. This system enables good artistic control but using this method for visual speech is problematic.

In visual speech we split a speech signal into atomic units called phonemes. Each phoneme is matched to a corresponding viseme, where a viseme is the extreme of mouth motion for a particular phoneme. Transitions between visemes then produce animation which is played in parallel with the speech signal. However, this approach does not incorporate coarticulation which is the major issue in visual speech.

Coarticulation models context in speech movement. Thus the shape of the mouth for a particular phoneme is affected by preceding (e.g. boot vs beet) and following (e.g. stew) phonemes. Essentially a viseme has an effect over a duration of the animation – its static appearance belies its dynamic influence. We can handle coarticulation for the pose target approach by using dominance functions [Cohe93], or by using constraint-based animation [Edge04a]. Alternatively, we can once again make use of MoCap.

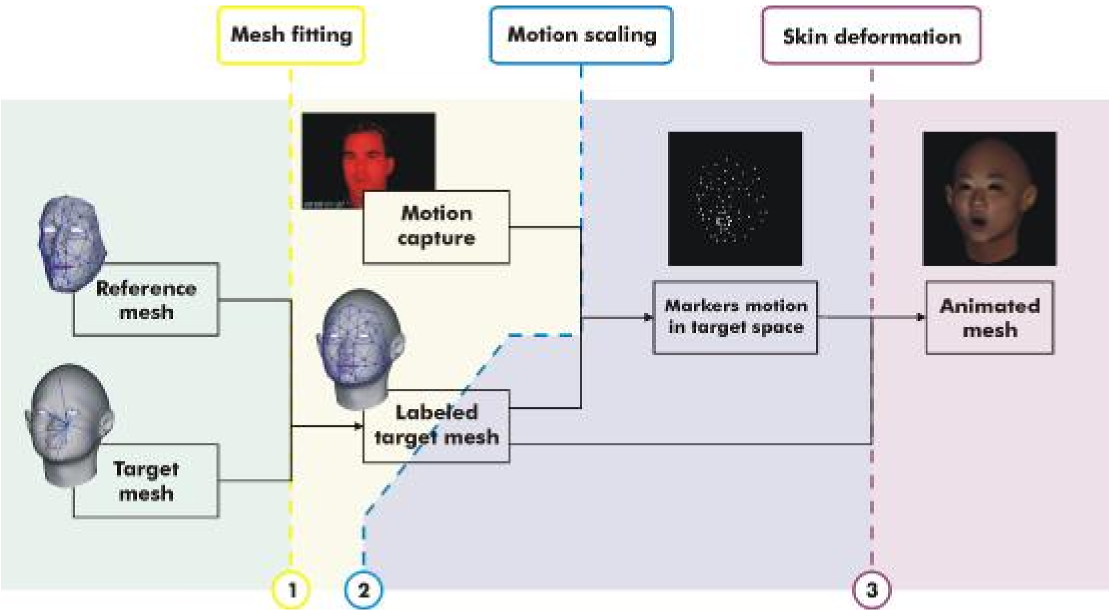

Figure 7: Using BIDS to animate a face from MoCap data.

The first stage in using MoCap is to capture a person speaking a vocabulary for the required application. Edge et al [Edge04b; Sanc03] use the example of a speaking clock where sentences are of the form: {prolog}/{time-info}/{day-info}. Examples would include ‘the time is now / exactly one / in the afternoon’ and ‘the time is now /

quarter to ten / in the morning'. This data is then segmented and labelled into syllables, words and sentences. With this information, new sentences within the limited domain can be constructed. Coarticulation is thus handled within individual dynamic units – the length of the MoCap fragments spans the duration of coarticulation effects.

The second stage is to use the captured motion to deform the target mesh. Animation control of a target facial mesh is achieved by using a surface-to-surface control technique called BIDS [Sanc04], which uses a Bezier-triangle surface to control a detailed facial mesh. The Bezier-triangle surface is formed by triangulating the set of MoCap markers. Figure 7 illustrates the complete animation process. First, the set of MoCap markers, which are defined on a reference mesh are mapped to the target mesh, i.e. the mesh that is to be animated. The labelled MoCap data (which is the same set of markers as defined on the reference mesh) is scaled to fit the target mesh. The target mesh is then animated by the MoCap data to produce the final animation. Figure 8 illustrates BIDS in action.

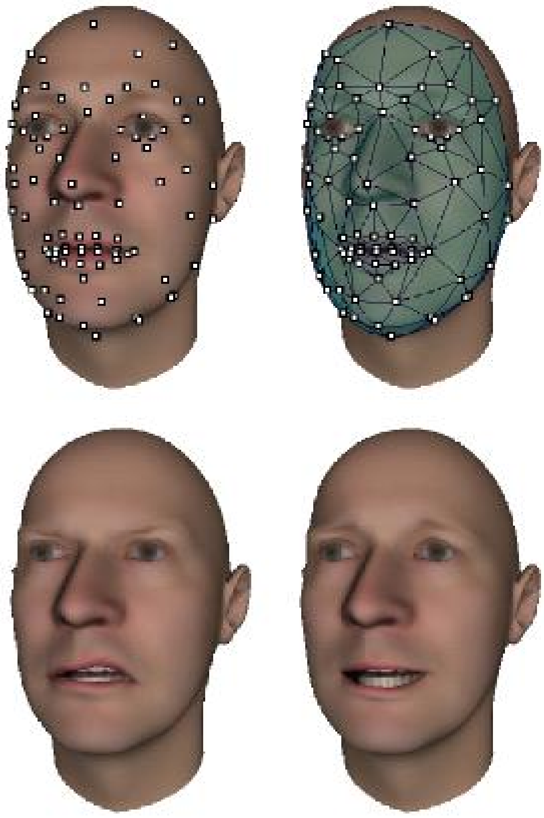

Figure 8: BIDS in action

This technique works well, but there are two limitations. First, coarticulation is handled within units, but not necessarily between them. For sentence-length units, visual speech will look very natural. However, this leads us to the second problem which is the size of the captured vocabulary. There would be an impossible number of sentences to capture for the English language. Capturing just words would itself be unfeasible, and so lower-level units must be used such as diphones (phoneme to phoneme transitions) of which there are approximately 1500 in English. But with shorter dynamic units, the issue of coarticulation once again becomes a problem. However, a limited vocabulary may not be such a problem in some games.

Using MoCap to drive facial animation suffers from the sparseness of data captured about the face. Since only the positions of a discrete set of markers are captured, the fine tissue detail, such as wrinkling, is not included. A layering approach can be used to include such detail. Sanchez et al [Sanc04] analyse the strain in a deforming surface in real-time and use a functional model of wrinkling to layer on a normal map. The functional model used is built using a pre-computed lighting analysis of a face undergoing a range of distortions. Figure 9 illustrates the strain analysis of two frames of MoCap applied to a target mesh.

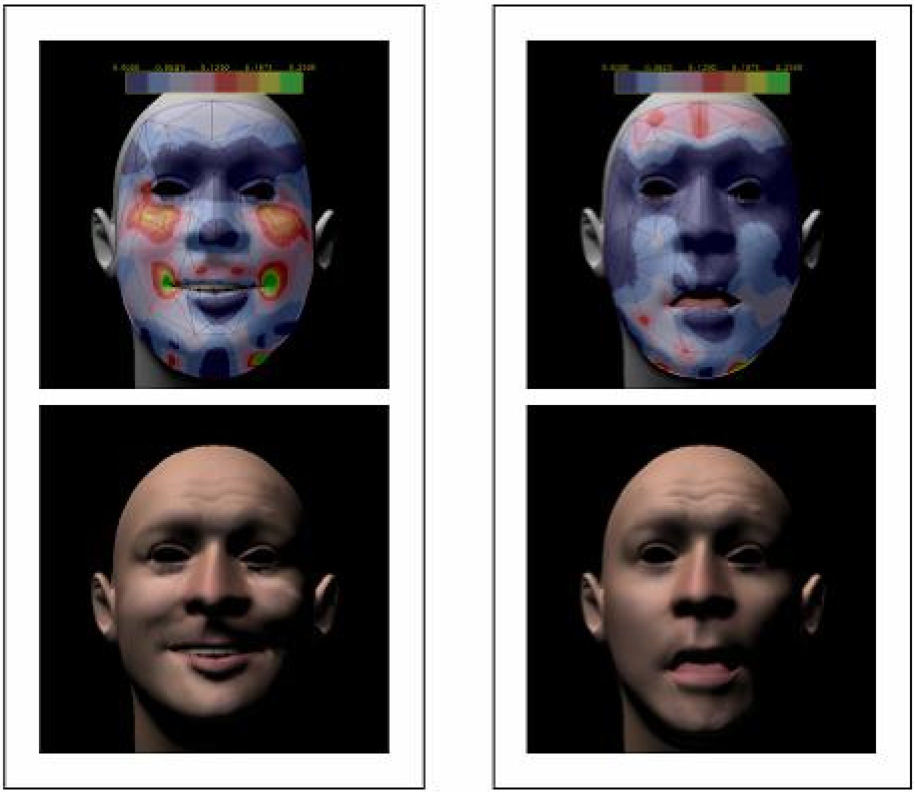

Figure 9: Strain analysis and small-scale deformations (wrinkling)

5 Conclusions

We have suggested that the development of more realistic characters in computer games is being held up by a reliance on the rather basic use of MoCap. Instead, such data should only be a starting point. We have described promising real-time techniques that can be used to build upon MoCap to offer more character realism, and to offer individualisation. With further work, communicating characters will appear that offer a game player a more human gaming experience. Perhaps we will see new game experiences that allow a player to act, sing or compete with famous people from down the ages, and the death of the FPS.

Acknowledgements

Thanks to our PhD students James Edge, Michael Meredith and Manuel Sanchez.

References

Banks94 Banks, D.C., Illumination in Diverse Codimensions, Proc. SIGGRAPH, 1994, 327-334.

Bees04 Beeson, C., Animation in the Dawn Demo, GPU Gems, Addison-Wesley 2004.

Cohe93 Cohen, M. and D. Massaro Modeling coarticulation in synthetic visual speech. In Proceedings Computer Animation ’93, 139–156.

Edge04a Edge, J. and S. Maddock, Constraint-based Synthesis of Visual Speech, ACM SIGGRAPH'04 Technical Sketch, 2004.

Edge04b Edge, J., M. Sanchez, S. Maddock, Using motion capture data to animate visual speech, Symposium on Language, Speech and Gesture for Expressive Characters, March 30-31, 2004, part of the AISB 2004 Convention: Motion, Emotion and Cognition, University of Leeds, UK, March 29 - April 1, 2004.

Gibb01 Gibbs, J., Rendering Skin and Hair, silicon-valley.siggraph.org/MeetingNotes/shrek/**hairskin**.pdf.

Gree04 Green, S., Real-Time Approximations to Subsurface Scattering, GPU Gems, Addison-Wesley 2004.

Hanr93 Hanrahan, P. and W. Kreuger, Reflections from Layered Surfaces due to Subsurface Scattering, Proc. SIGGRAPH 1993, 165-174.

Land99 Lander, J., Real-Time Cloth Animation, www.gamasutra.com/features/20000327/**lander**_01.htm.

Mere04 Meredith, M. and S. Maddock, Adapting Motion Capture using weighted Real-Time Inverse Kinematics, Proc. GDTW 2004.

Mert04 Mertens, T., J. Kautz, P. Baekert and F. Reeth, A Self-Shadow Algorithm for Dynamic Hair using Density Clustering, Eurographics Symposium on Rendering, 2004.

Mori70 Mori M. (1970). Bukimi no tani [The uncanny valley]. Energy, 7(4), pp 33-35. (see ‘MacDorman, K.F. (2005). Androids as an Experimental Apparatus: Why is there an uncanny valley and can we exploit it? Proc CogSci-2005 Workshop: Toward Social Mechanisms of Android Science, 25-26 July 2005 in Stresa, Italy, pp. 106-118’ for a translation)

Oshi01 Oshita, M., Real-time Cloth Simulation with Sparse Particles and Curved Faces, SIGGRAPH 2001, Sketches and Applications.

Sanc03 Sanchez Lorenzo, M., J.D. Edge, S. King, and S. Maddock, Use and Re-use of Facial Motion Capture Data, Proc. Vision, Video and Graphics 2003, University of Bath, July 10-11, 2003, pp. 135-142.

Sanc04 Sanchez, M., J.D.Edge, S.C.Maddock, Realistic Performance-driven Facial Animation using Hardware Acceleration, Department of Computer Science Research Memorandum CS-04-10, University of Sheffield.

Shin01 Shin, H.J., J, Lee, M. Gleicher and S.Y. Shin, Computer Pupputry: An importance Based Approach,ACM Trans. on Graphics, 20 (2), 67-94.

Watt92 Watt, A. and M. Watt, Advanced Animation and Rendering Techniques: Theory and Practice, Addison-Wesley, 1992.