LoD Management on Animating Face Models

Hyewon Seo, Nadia Magnenat Thalmann
MIRALab, University of Geneva
24 rue du General Dufour
CH-1211 Geneva, Switzerland
{seo, thalmann}@cui.unige.ch

Abstract

In this paper, we present our work on a level of detail(LoD) technique for human-like face models in virtual environments. Conventional LoD techniques have been adapted to allow facial animation on simplified geometric models. This includes optimization on both geometry, and animation parameters. Simplified models are generated in a region-based manner in consideration with the mobility of each region. The animation process is decomposed into two sub-processes and each step is optimized. In the MPA(Minimum Perceptible Action) level optimization, a hierarchical structure was devised for the multi-level animation model. The deformation level is simplified by reducing the number of control points. During runtime, the level of animation is selected in combination with the viewpoint information the level of geometry.

Keywords : *levels of detail, virtual humans, real-time facial animation.*

1. Introduction

Many attempts have been made to bring realism into virtual environments. Realistic virtual humans, either autonomous or representative, play an important role in giving the feeling of presence and realism to the users[1][2]. However, they generate problems with real-time animation because of two main reasons. Firstly, they are represented by highly detailed geometric models, which typically means a polygonal mesh composed of thousands of textured triangles. Despite the enormously improved performance of graphics accelerators, it is often the case that the scene contains more triangles than can be rendered at interactive frame rates as the number of virtual humans increases. Secondly, in order to simulate realistic movements, they are driven by a number of animation engines such as face and body animation, each of which involves heavy computation during run-time.

During the last ten years we have made considerable efforts to develop a real-time animation system[2]. However, in a scene where a number of such virtual humans exist, the rendering task becomes the bottleneck of the real-time performance. Obviously, improving it in such a way that it is possible to render a group of virtual humans with real-time performance remains a challenge.

To improve rendering performance, it is usual to define several versions of a model at various levels of geometric detail. A detailed model is used to represent a nearby object while a coarser one is used for a distant object from the viewpoint.

In this paper, we describe a LoD technique applicable for animating virtual human face models. The basic idea is more or less the same as for other objects in virtual environments: A detailed description of a face is necessary only when it's close to the viewpoint. Here, the description includes both geometry and animation. Our method starts with a multi-resolution geometric model generation. Considering several features of a face model, a region-based approach is used for generating a number of models of different resolutions. The region here is a basic unit of muscle action simulation in animation. A number of regions are grouped together and different importance weights are used for the decimation of each region group considering several observations on face models. Then a hierarchical structure, named as 'active tree', is constructed to incorporate an efficient animation parameter control method during run-time. The level transition of the animation is transformed to either a shrink or an expansion of this tree wherein a set of active regions and animation parameters are kept. Finally, a proper level of deformation is used depending on the current level of the geometric model.

This paper is organized as follows: In the following section, we explore related works for LoD generation and control on animating objects or virtual humans. Our

virtual human face models and the animation are described in Section 3. Section 4 details our region-based approach to build multi-resolution models and how the active tree works to manage facial animation on simplified models during run-time. Results as well as implementation issues are presented in Section 5. Finally we conclude with suggestions for future work in Section 6.

2. Previous work

Although the idea of muti-resolution models is quite old, there are not many results for animating objects, particularly for virtual humans. Most of them focus on static objects, and rarely simply moving objects. Lau et al. have presented a multi-resolution method that generates run-time viewing and animation parameters in [3]. The issues of level of detail on human models have been addressed in [4], although they do not deal with facial animation. Their approach is based on the generation of multiple definition of a human model and the simple animation frame skipping.

The issue of applying LoD techniques to VR environments involving deformable objects has been addressed in [5] along with their experiments on deformable hands. In their work, the deformation, which is considered to be part of the rendering pipeline, is divided into two main processes and each process is optimized to reduce the calculation time. In the first process, where skeleton animation is transformed to the control points movement, interpolation between the two extreme positions and orientations are used to replace point-based update. In the second step where control lattice are used to deform a sets of points within their affecting range, the degree of the deformation function as well as the number of control points has been reduced to get optimized results. However, this method only focused on the deformation process and not the whole pipeline. Their method did not consider other important factors of LoD - geometry and viewpoint information.

Capin et al. have introduced a dead-reckoning technique for articulated virtual human figures to decrease the number of messages in a networked virtual reality system[6]. A dead-reckoning algorithm based on Kalmann filtering has been applied to transmit joint-angles of body models through network.

Carlson and Hodgins in [7] introduced their simulation levels of detail for real-time animation of legged creatures in a virtual environment. They construct three levels of detail in their multiresolution models : rigid-body dynamics, a hybrid kinematic/dynamic representation, and a point-mass simulation. However, they used quite simple geometric models for the rigid body representation and did not try to combine geometric LoD with simulation LoD.

3. Virtual human face models

3.1 Face model

With our previously developed method to reconstruct a face model from two orthogonal images[8], we could simulate populated virtual environments with realistic virtual human figures. The method uses Dirichlet Free-Form Deformations(DFFDs)[9] to deform predefined feature points on the generic model as control points, and then other points correspondingly. Feature points are located on the input images in a semi automatic manner, as shown in Figure1.

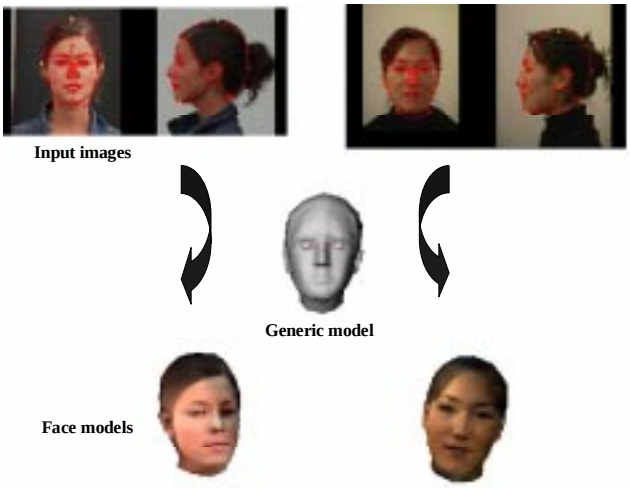


Figure 1. Generic model and cloned virtual human face model (2313 textured polygons each)

Our generic model is composed of 2313 triangles including teeth and eyeballs for the animation afterwards. The underlying animation structure describes about 20 regions defined on the mesh. The region, defined as a set of adjacent polygons, is a basic unit for simulation of muscle movements (Figure 2). Note that all heads from this method have the same topology. Moreover, as model fitting transforms the generic model without changing the underlying region structure, they share the same animation structure.

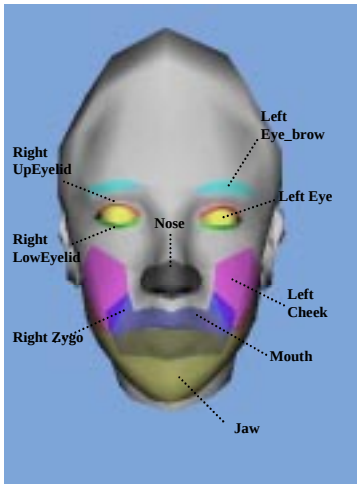


Figure 2. Regions for facial animation

Although this reconstruction method is highly efficient to model and simulate realistic human faces, the resulting meshes are seldom optimized for rendering efficiency, as can be seen in Figure 1. As a result, it is expensive to store, transmit and render them.

3.2 Facial animation

3.2.1 Multi-level Control

Our facial animation module[11] provides several levels of control (Figure3). On the lowest level, we use a set of 65 Minimum Perceptible Actions or MPAs, each of which describes a muscle movement of a region on the face. These muscle actions are simulated by applying Rational Free-Form Deformations(RFFDs) on the regions.

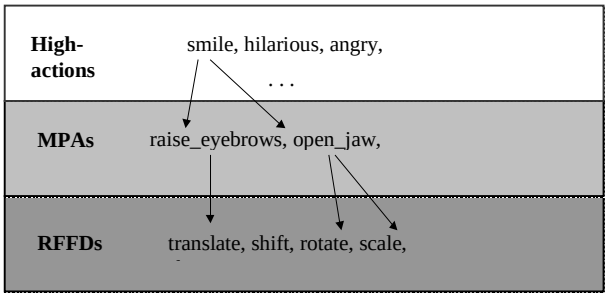


Figure 3. Three levels of control in the animation system

A number of MPA values in Figure4 show how a 'smile' expression is transformed into MPA level.

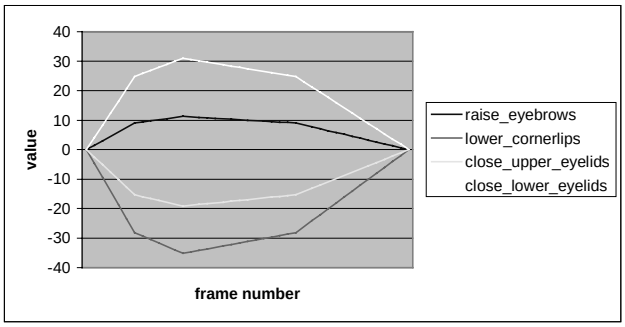


Figure 4. Selected MPAs for 'smile' expression

3.2.2 Deformation of the model

FFD (Free form deformation) is a technique for deforming solid geometric models in a free form manner[10]. Given a surface primitive of any type or degree, it works on an imaginary parallelepiped of flexible plastic that embeds the target primitive. It is deformed along with the deformation of the surrounding plastic.

One of the critical limitations of the FFD model is its use of rectangular arrangement of control points. Many extensions to overcome this limitation have been proposed, among which are EFFD[12], NFFD[13] and DFFD[9]. The most general of all these is the DFFD where any constraint on the position and topology of control points are removed. In this approach, any point of the surface to deform located in the convex hull of a set of control points in general position, is expressed relative to a set of the control points set with the Sibson coordinate system.

Rational Free Form Deformation (RFFD) is an extension of DFFD to give one more degree of freedom to it. As in [14], RFFD assigns weights to the control points. In this context, DFFD is a special case of RFFD where all control points are given with the same weights.

In our face model, $3 \times 3 \times 3 = 27$ control points are defined along the bounding box of each region. An MPA is implemented by a set of deformations of the selected regions. The translation of the eyebrow region in Figure 5, 'raise_eyebrow' MPA. for example, is one of the deformations necessary for the 'raise_eyebrow' MPA.

Figure5. Implementation of 'squeez_eyebrow' MPA by applying RFFD on the eyebrow region. Control points are shown as white lattice.

4. LoD on virtual human faces

The LoD technique presented in this paper was conceived for animating face models in VR environments, for which existing LoD methods are inadequate.

There are several important points to be considered : Firstly, the region based animation should be applicable for simplified models. This requires the animation structure including region information to be constructed through a simplification process. Secondly, preprocessing for the multi-resolution modeling should consider both the animation parameters and geometric description of the model. Finally, LoD control during run-time should incorporate the viewpoint information, animation parameters, and the geometry.

Figure 6 shows the relationship among these three components. The viewpoint information is used to control geometric level as well as that of the animation parameters. The level of the animation parameters is dependent on the geometrical description of the corresponding region in the sense that the number of control points is reduced with the simplified geometry. The level of geometry is affected by the animation components because each region is assigned with different importance weights depending on its mobility. As a result, highly mobile regions such as the eyebrows and the lips are given with higher weighting factors than the teeth or the ears regions.

In the remainder of this section, we describe how these components are implemented in consideration of to develop our LoD method adequate for face models.

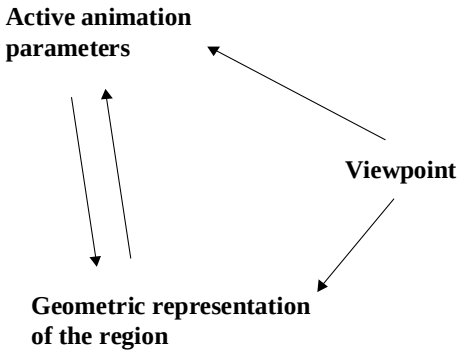


Figure6. Three main components and their dependencies in the LoD system

4.1 Geometric LoD

From conventional LoD research, sharp edges, high curvatures, and silhouettes have been regarded as important features. Many simplification methods focus on preserving these prominent features during the simplification process[15][16]. On face models, however, geometric importance is not the only factor for visual importance.

It has long been observed that faces are 'special' objects in the visual world of the human[17]. For example, the human face is highly mobile, and the movement of it is important for interpersonal communication as well as recognition. This means that the moving parts of the face also contribute to its visual importance. Moreover, other attributes such as color or texture associated with vertices give rise to discontinuities in the visual appearance of the face model.

With this in mind, we learned several lessons to generate the next LoD model from the original model.

- Even though ears, for instance, have many high curvatures occupying quite a part of the triangles in the original model, they seldom attract visual attention, remaining static all the time. It is not so useful to keep these geometric features in simplified models.
- Eyeballs are also represented with semi-spheres in order to allow simulation of eyeball movements during animation. However, they are too small to be visible when viewed from a distance. Eyeball movements and their detailed geometric description can be dramatically discarded as the level increases.
- On the other hand, lips and eyebrows move frequently and thus form visually noticeable

features during runtime. These should be kept even in the simplified description of the face model, although they may not be geometrically important.

4.1.1 Non-uniform importance weight distribution - Region based approach

Based on previous observations, we are motivated to assign importance weights to the original face model in a non-uniform way. This importance weight is used to decide which vertices to remove during the generation of the next LoD model. Visually unimportant, yet triangle budget-consuming regions such as ears, teeth, and eyeballs can be efficiently dealt with by assigning them the lowest weight.

We have implemented this non-uniform distribution of importance weight using existing animation regions in Figure 3 as basic units. These regions give a natural division of the face model in many aspects – They are not only basic units for animation but also give a good approximation of color or texture based subdivision. However, some regions are complicated since they are overlapping and thus simplification of one region affects the other. Cheek and zygo regions can be examples. This problem can be simplified by identifying independent groups of regions. Face mask, ears, eyes, and teeth region groups were chosen.

LoD generation is started with these four region groups and predefined importance weights on them. Regions with the same importance weight are grouped together to be dealt with by the same simplifier. Within a group, the simplifier traverses the geometry set and removes vertices until a specified percentage of the original number of triangles remains or the removal of vertices is no longer possible due to other criteria.

We use one of the derived classes of opSimplify class from OpenGL Optimizer library[18] for the simplifier. The evaluation function of the simplifier is

$$\text{Eval(vertex)} = W_r \times (W_0 \times \text{distance} + W_1 \times \text{normalDeviation} + W_2 \times \text{curvature})$$

where W_r is the importance weight of the region, W_0 is for the distance between the old vertex and the average plane of the simplified polygon, W_1 for the normal deviations of a vertex and thus sharp features of the mesh, and W_2 for high curvature regions. Each of the weights, when it has a high value compared to the other weights, preserves different characteristics of the mesh[]. As can be easily guessed, it is now possible for the eyes, teeth and ears to be simplified dramatically and finally to vanish by adjusting importance weights.

4.2 Animation LoD

As mentioned in section 3.2, the animation process can be decomposed into two subprocesses : one in the MPA level and the other in the deformation level. Each process can be optimized to save computation time.

4.2.1 MPA level LoD - Active tree structure

The only optimization is to reduce the number of MPA parameters. A hierarchical structure named as 'active tree' in Figure 7 describes our method to manage active regions and animation parameters defined on them. An active tree is composed of more than one level of the hierarchy. Each level in the hierarchy includes a set of regions. The higher the level a region belongs to, the more mobile and/or visually important its movement is.

At any moment, the level of the active tree is selected depending on the viewpoint information. Those regions inside the active tree is defined to be active, and animation parameters concerning this region are processed. Figure 7 shows the active tree of level 2 where only the lower-mask group is active. Whenever there is a level transition, it either shrinks or expands to maintain a proper set of active regions. The active tree of the highest level contains no region group. This reflects the observation that when the model is far enough, no animation is needed.

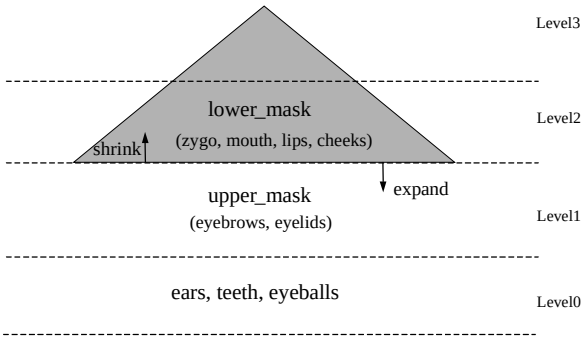


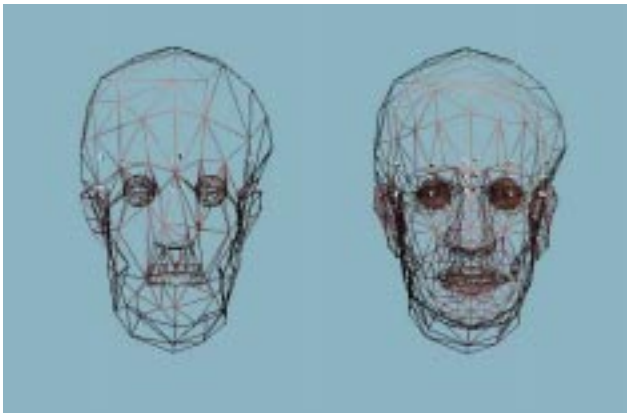
Figure 7. Hierarchical region management : active tree for level 2 model

4.2.2 Deformation level LoD

Deformation of a region is done by first moving control points of that region, and then calculating the movements of each points of the enclosed surface. Although this process is sped up with the simplified geometry of the enclosed surface, it is still possible to define multiple levels of deformation by reducing the number of control points. In Figure 8, the control lattice of

dimension $3 \times 3 \times 3$ (27 controls) are reduced to $2 \times 2 \times 2$ (8 controls) on the eyebrow region.

(a)



(b)

Figure 8. Use of different number of control points for different levels of geometric models. 8 points are used on the left and 27 points on the right.

4.2.3 Frame skipping

Instead of playing an animation frame by frame, frame skipping can be used to speed up animation[4]. We extended this method to vary the frame skipping rate according to view distance from the model. Close faces are given with higher frame update rates than further ones. This optimizes the visual effect with the same number of frames skipped. However this method has drawbacks because the duration of the animation sequence varies depending on the location of the face model.

5. Results

Our LoD method was implemented and tested on a personal computer with Elsa Gloria-XL graphics adapter (16MB memory). The C++ language with OpenGL Optimizer and Cosmo3D graphics library has been used for development. Table 1 shows the average rendering time and maximum per-frame animation times for various levels of detail of a face model. As the level increases, the animation time as well as the rendering time is reduced. At the lowest resolution model (level 3), all animation parameters are skipped, resulting in no CPU time spent for calculation of animation.

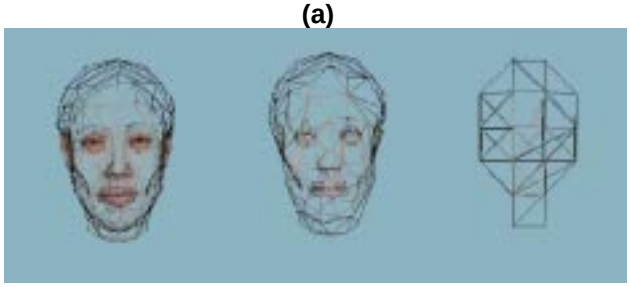
Table 1. Rendering and animation time per frame for different levels of a model

	Number of triangles	Number of vertices	Average rendering time (ms)	MAT (ms, Exp.1)	MAT (ms, Exp.2)
Level0	2313	2230	23.39	13.16	13.04
Level1	1533	1478	11.83	3.84	0.85
Level2	357	348	7.78	0.0	0.0

(MAT : Maximum Animation Time, Exp : Expression)

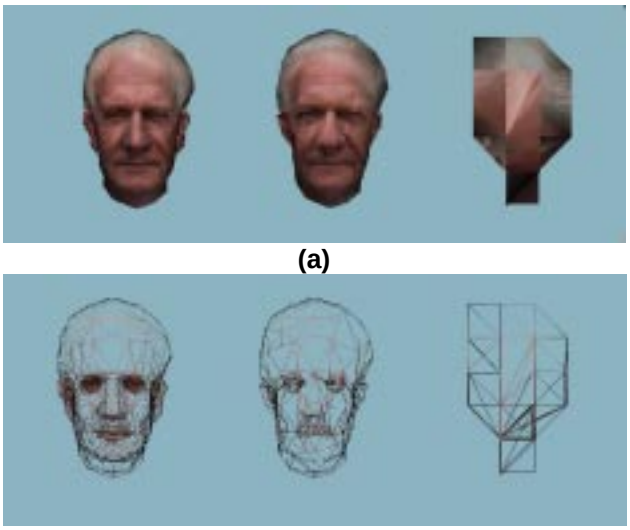
Figure 9~Figure 10 show three different levels of various face models with several expressions. As can be seen, level 1 models, which are composed of approximately 50% of the original models, do not bring much difference to the visual appearance. Face models with different view distances are shown in Figure12.

(a)



(b)

Figure 9. Different levels of a young woman's face model. (a) Gouraud shading (b) wireframe



(a)

(b)

Figure 10. Different levels of an old man's face model. (a) Gouraud shading (b) wireframe

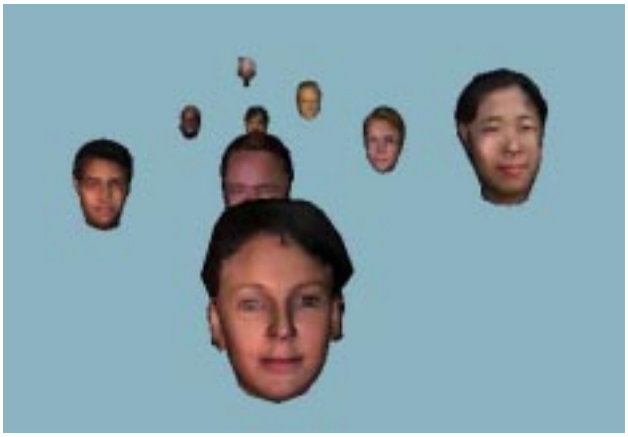
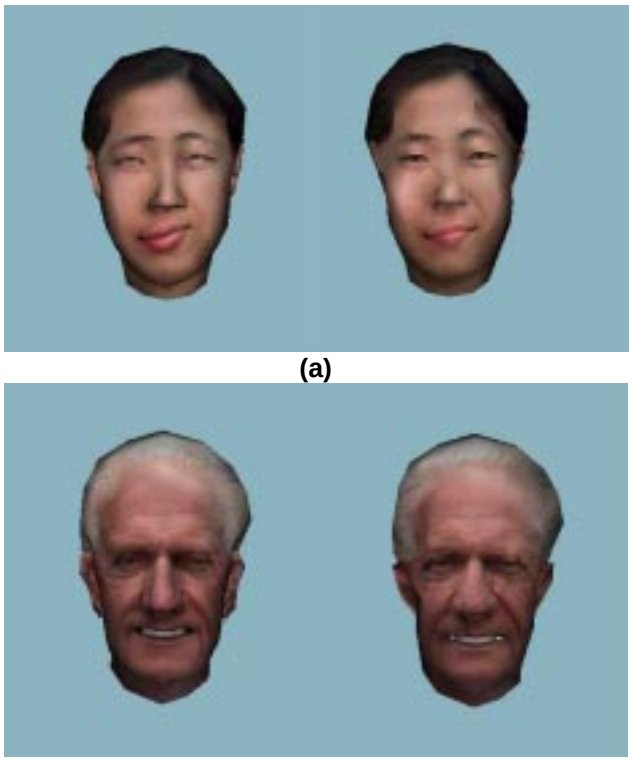


Figure 11. A VR environment populated with face models

Animations applied to different levels of the model are shown in Figure12. Note that only the eyebrows and the lips regions are active in level 1 models. As a result, the animation becomes less expressive because neighboring regions of the lips such as the cheeks and the zygos are not activated. For the eyebrows animation in level 1 modles, 8 control points are used instead of 27.



(a)

(b)

Figure 12. Expressions applied to different resolution models (a) young woman's case (b) old man's case

6. Conclusions and Future works

We have shown in this paper our framework to handle the geometric LoD in combination with animation LoD to maintain a real-time facial animation of realistic face models in virtual environment.

The main result of our work is simplified geometric models together with simplified animations. Multiresolution model of the geometry was done in consideration with not only its geometric features such as sharp edges and normal deviations, but also its mobility. For the animation a hierarchical structure is developed, which naturally supports the animation parameters culling and offers a clear description of the idea of LoD control during runtime. Finally, the optimization of deformation was implemented by reducing the number of control points. When the level of geometry is lower than that of animation, the animation level is adaptively lowered by using the smaller number of control points.

Our method does not solve all problems. Firstly the animation LoD method may not applicable for all animation or deformation models. For example, a DFFD-

based facial animation does not come along with region information since it does not require region information. Moreover, it is not fully automatic in the sense that the weighting factor was assigned in an ad hoc manner. Also in the deformation LoD may vary depending on the deformation models used for animation.

Our future works will include the following:

- Support for standards (MPEG-4) : Our work can find a useful application with facial animation in MPEG-4. By reducing the number of Facial Animation Parameters(FAPs)[19], the network load will be efficiently lightened.
- Adaptive LoD based extension : Instead of discrete LoD used, adaptive or continuous LoD generation and control method can be used to improve the visual quality. Also view orientation information along with the view distance information can be considered.
- Extension to body model and animation : The similar method including the active tree structure can also be applied to implement LoD on bodies.

Acknowledgements

The authors would like to thank Wonsook Lee for providing face models and Prem Kalra for his helpful comments on this research. We also thank Prithwesh De for his proof reading this paper.

This work is partially supported by the European Research Project Virtual Amusement Park (VPARK).

References

[1] Pandzic,I., Capin,T.K., Magnenat-Thalmann,N. and Thalmann,D., "VLNET: A Body-Centered Networked Virtual Environment", *Presence: Teleoperators and Virtual Environments*, 6(6), pp.676-686, MIT Press, 1997.

[2] Kalra,P., Magnenat-Thalmann, N., Moccozet,L. and Sannier,G., "Real-time animation of realistic virtual humans", *IEEE Computer Graphics and Applications*, 18(5), IEEE Press, 1998.

[3] Lau,W.H., To,S.P. and Green,M., "Adaptive Multi-Resolution Modeling Technique Based on Viewing and Animation Parameters", *Virtual Reality Annual International Symposium (VRAIS '97)*, pp.20-27, IEEE Press, 1997.

[4] Musse,S.R., Babski,C., Capin,T. and Thalmann,D., "Crowd Modelling in Collaborative Virtual Environments", *Virtual Reality Software and Technology (Proc. VRST '98)*, pp.115-124, ACM Press, 1998.

[5] Moccozet, L. and Magnenat-Thalmann,N. "Multilevel Deformation Model Applied to Hand Simulation for Virtual Actors", *Proc. VSMM '97*, pp.119-128, IEEE Press, 1997.

[6] Capin,T.K., Pandzic,I.S., Magnenat-Thalmann,N. and Thalmann,D., "A Dead-Reckoning Algorithm for Virtual Human Figures", *Virtual Reality Annual International Symposium (VRAIS '97)*, pp.161-169, IEEE Press, 1997.

[7] Carlson D.A. and Hodgines J.K., "Simulation Levels of Detail for Real-time Animation", *Proc. Graphics Interface '97*, pp.1-8, 1997.

[8] Lee,W. and Magnenat-Thalmann,N., "Head Modeling from Pictures and Morphing in 3D with Image Metamorphosis based on triangulation", *Proc. Captech98 (Modelling and Motion Capture Techniques for Virtual Environments)*, pp.254-367, IEEE Press, 1999.

[9] Hsu W., Hugues J. F., Kaufman H., Direct Manipulation of Free-Form Deformations, *Proc. SIGGRAPH'92*, pp. 177-184, ACM Press, 1992.

[10] Sederberg T.W. and Parry S.R., "Free-Form Deformation of Solid Geometric Models", *Proc. SIGGRAPH'86*, pp.151-160, ACM Press, 1986.

[11] Kalra,P., Mangili,A., Thalmann N.M. and Thalmann,D., "SMILE : A Multilayered Facial Animation System", *Proc. IFIP Conference on Modelling in Computer Graphics*, pp.189-198, Springer, 1991.

[12] Coquillart S., Extended Free-Form Deformation : A Sculpting Tools for 3D Geometric Modeling, *Proc. SIGGRAPH'90*, pp. 187-196, ACM Press, 1990.

[13] Lamousin H. J., Waggenspack W. N., Nurbs-based Free-Form Deformations, *IEEE Computer Graphics and Applications*, 14, 16, pp 59-65, IEEE Press, 1994.

[14] Kalra P., Mangili A., Magnenat-Thalmann N., Thalmann D., Simulation of Facial Muscle Actions Based on Rational Free-Form Deformations, *Computer Graphics Forum*, 2(3), pp. 65-69, Blackwell Publishers, 1992.

[15] Schroeder,W., Zarge, J. and Lorensen,W., "Decimation of Triangle Meshes", *Computer Graphics*, Volume 25, No. 3, *Proc. SIGGRAPH '92*, pp.65-70, ACM Press, 1992.

[16] Cohen,J. Varshney,A., Manocha,D., Turk,G. and Weber,H., "Simplification Envelopes", *Proc. SIGGRAPH '96*, pp.119-128, ACM Press, 1996.

[17] Vicki,B., Hancock,P. and Burton,A.M., "Human face Perception and Identification", *Face Recognition : from Theory to Applications*, Vol.163, pp.51-72, Springer, 1998.

[18] Wennerberg L. "Chapter6 : Rendering Appropriate Levels of Detail", *OpenGL Optimizer Programmer's Guide : An Open API for Large-Model Visualization*, 1997.

[19] Koenen,R. "Coding of Moving Pictures and Audio", *International Organisation for Standardisation ISO/IEC JTC1/SC29/WG11*,http://drogo.cselt.stet.it/mpeg/standards/mpeg-4/mpeg-4.htm, 1999.