ALOHA : Adaptive Level Of Detail for Human Animation

Towards a new framework

Thanh Giang, Robert Mooney, Christopher Peters and Carol O Sullivan

Image Synthesis Group, Trinity College, Dublin 2, Republic of Ireland

Abstract
The task of animating and rendering virtual humans in real-time is challenging. One must first establish a sense of realism through appearance, and then maintain this realism through correct and plausible motion, while continually satisfying the real-time constraint imposed. One technique that aids in achieving such a task is to take advantage of the viewer's perception to compute less accurate models when loss of accuracy is unlikely to be noticed. Traditionally, these 'level of detail' techniques have been used primarily for geometric modelling. However, more recently, they have also been applied to animation. This paper seeks to amalgamate animation and geometric level of detail research in order to produce an integrated framework on which to build a totally scalable system for virtual human animation.

Keywords: level of detail, virtual humans, real-time simulation

1. Introduction

Despite the increase in computing power and the documentation of a large number of different character animation techniques, a number of factors are still found to be lacking in many real-time character animation systems. Perhaps the most noticeable of these factors is scalability. The computational cost of displaying and animating a number of articulated, multi-layered characters can quickly mount.

A number of solutions exist for alleviating this problem. The first involves distributing simulations among many processors [9]. The second approach involves the use of a number of differing simulation or rendering models, each with an increased degree of computational complexity. It is this 'level of detail' approach that has been adopted for the ALOHA framework.

2. Previous Work

The term 'level-of-detail' has been widely used in relation to research on levels of detail in geometric models [7]. As Carlson and Hodgins point out, these geometric techniques are also relevant to systems that vary the detail of simulation [3]. Essentially, this means that one may switch between animation levels of varying computation complexity according to some set of predefined rules. An important criterion for such rules is that viewer perception of any degradation in the simulation remains minimal.

Carlson and Hodgins have presented a method for animating one-legged creatures at multiple levels of detail depending on the importance of a creature and its visibility. Simulation methods include rigid-body dynamics and hybrid kinematics / dynamics, depending on the accuracy required. Cozot et al have suggested a level of detail architecture consisting of a pipeline of sub-models, where each sub-model performs a given task in the animation process. The architecture is applied to a walking model for complex grounds. Level of detail is selected according to two criteria: the complexity of the environment and the distance from the camera [5].

3. ALOHA System Overview

Thus far, level of detail research has focused on either pure geometric or pure animation levels of detail but never both together within the one system. We propose a viable framework that will amalgamate both research areas into the one scalable system for the purpose of virtual human animations. We require a system to be robust enough to deal with realistic real time characters in a dynamically changing environment as well as keeping a consistently high and smooth frame rate. Switching between the different levels of details during our simulations should be smooth and unnoticeable. As well as being robust, our system should be scalable and easily upgradeable for future additions.

A high level view of the ALOHA framework reveals four major modules *(Figure 1)*, the knowledge base, LOD resolver, motion controller and geometric controller. Each module has been designed so that it is almost completely self-contained so that future enhancements can be easily adopted.

Knowledge Base: ALOHA's knowledge base is fundamentally a tabulated set of rules that controls the level of detail for a given object / scene. The rule set can be tuned by the user to be in accordance with the capabilities of the underlying hardware. Rules are specified to control each object's geometric and animation level of detail based on simple criteria i.e. velocity, position in the viewing plane, distance from the viewer etc.

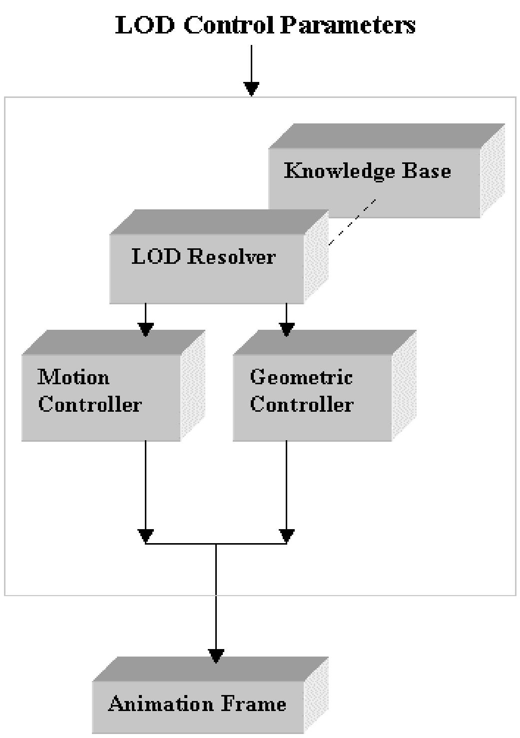

Figure 1: *High level view of the ALOHA framework.*

LOD Resolver: This module lies at the heart of the ALOHA framework. The LOD resolver not only controls switches between levels of detail, but it also marshals the parameters that guide the geometric and motion controllers. Communication between the LOD resolver and both controllers occurs on a master/slave basis. After extracting the necessary object attributes from a frame/scene the LOD resolver queries the knowledge base and shunts the results to the controller(s). Furthermore, this module acts as a layer of abstraction between scene control and scene composition, which allows for a more expandable system.

Motion Controller: The motion controller is responsible for handling and storing any animations that are required. It is the duty of the motion controller to register animation requests, to store necessary animation information, to update current animations and to manage the storage of state vectors for objects. The motion controller has the ability to schedule any requested animation at any level of detail. The actual level of detail is specified by the LOD resolver.

Geometric Controller: The geometric controller encapsulates all the information necessary to describe the object's appearance within the current frame. Currently the level of detail for the model's outer layers (muscle, fat and skin) can be specified by a single parameter that controls the geometric tessellation. Peer to peer communication occurs between both controllers. A state vector that describes the joint orientations is obtained from the motion controller. This vector is then wrapped in the outer surfaces to the level of geometric smoothness required.

For every object (typically a virtual human) within the scene that exhibits motion the LOD resolver must call the motion controller in order to resolve the motion. The LOD resolver does not bother calling the motion controller for objects within the scene in a rest state (i.e. exhibiting no motion). This ensures that we do not do any needless computations within scenes that may contain many dynamic objects but only one of which is asserting any motion while the others are at rest. Not only would this be too computationally expensive for slower processors to handle, it is also needless and wasteful. However, the geometric controller is called per frame because potentially the tessellation within the scene can change from frame to frame.

Our animation controller supports an initialisation module, which initialises all necessary data for the animation method chosen by the LOD resolver. It also provides a motion execution unit which incorporates an 'animation cache' to store the necessary data for each object of our animation and several sub modules which are called for the various animation production methods (i.e. a kinematics, a dynamic and a motion capture sub module etc). This motion control architecture is illustrated in Figure 2.

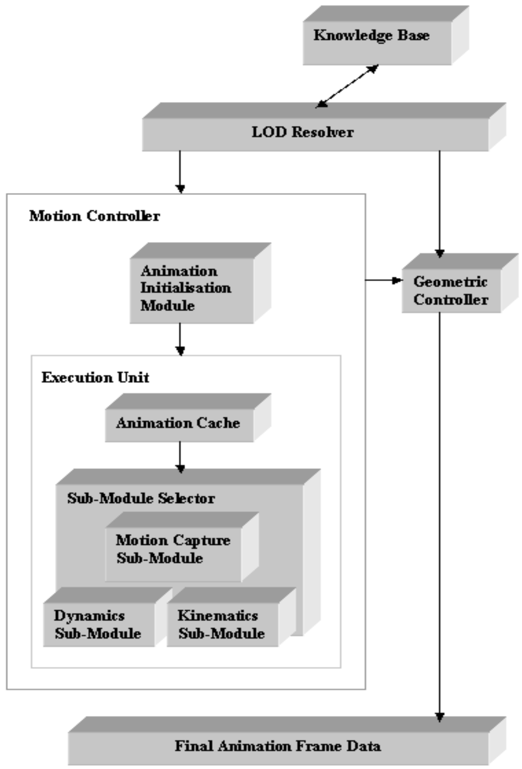

Figure 2: *Low level view of the ALOHA framework.*

4. ALOHA's Layered Model

Chadwick & Haumann first proposed a layered approach for developing animated characters in order to create perceptually realistic character deformations during motion [4]. Each layer exploits the functionality of the layer below, while exporting its own to the one above. In this way only the innermost layer (the skeleton) need be scripted for animation in order to achieve consistent deformations. Typically, as the number of layers increase, so too does the computational complexity of the animation. This ultimately prevents real-time simulation.

Since the ALOHA framework describes a four layered model (skeleton, muscle, fat and skin), geometric optimisations must be made in order to allow real-time animation in scenes of high complexity.

The skeletal layer represents the true 'bones' of the animation system. The skeleton itself is an articulated structure, consisting of a series of interconnected joints, each of which must observe a number of constraints. The collective position and orientation of these joints represents a single pose or 'state' that the skeleton is in. The main character animation algorithms act only on the skeleton.

The muscle layer is implemented by preserving the volume of the muscles' primitives, for which we use ellipsoids while the fat layer is modelled using stiff linear springs to preserve the distance between the muscles and the skin. The skin layer is simply a mesh of points constructed from the outer spring ends used in the fat layer (Figure. 3).

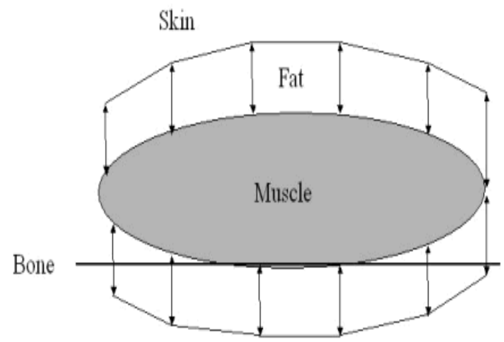

Figure 3: *ALOHA's 4-Layered Model*

4.1 Character Deformation

The extent of the surface deformation is dependent on the position of the local joints, while feasible muscle shapes are maintained using simple anatomical principles as outlined by Wilhelms [1,4] and Scheepers et al [8].

Muscles are connected across joints by two tendons. The tendon nearest the spine is referred to as the origin, while the other is referred to as the insertion. Each tendon is comprised of inelastic material that is anchored to a single bone, local to the joint.

Since the geometric primitive used to model muscles is the super-ellipsoid, first introduced by Barr [1], it is necessary to preserve the primitive's volume[14,8]. The surface of a super-ellipsoid can be described by a set of points *F*, where each coordinate *(x,y,z)* is determined from :

$$x = radius_x * \cos(phi)^A * \cos(theta)^B \quad (1)$$
$$y = radius_y * \cos(phi)^A * \sin(theta)^B \quad (2)$$
$$z = radius_z * \sin(phi)^A \quad (3)$$

where $-\pi < theta < \pi$, $-\pi/2 < phi < \pi/2$.

An ellipsoid can be defined using the equations (1-3) above by setting both A and B to 1 and specifying the ratios between the radii in the x, y and z axes. For example setting the ratios x:y:z to 1:1:1 will give a sphere while 2:1:1 will give a useful ellipsoid shape. Changing the A and B parameters values results in other super-ellipsoid shapes being formed (e.g. cylinders, cones and cubes). Recently, Sinnott and Howard [10] used this variety of shapes to implement virtual environments from captured real-world scenes.

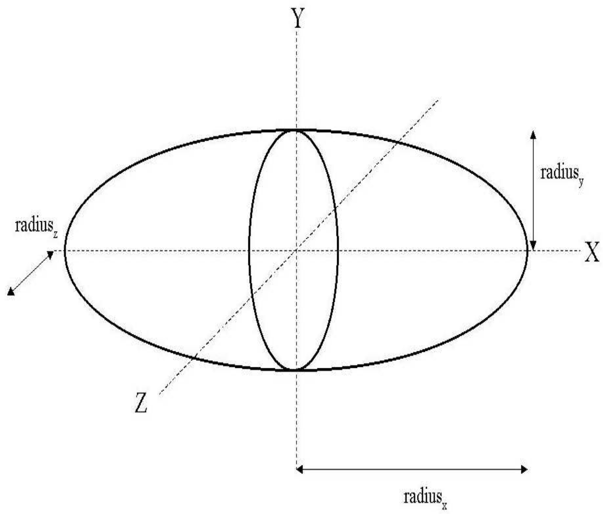

Figure 4: *Ellipsoid radii definition*

Since the volume of an ellipsoid is given by the formula

$$v_{ellip} = (4\pi/3) * (radius_x\, radius_y\, radius_z) \qquad (4)$$

volume preservation can be achieved once the ratio *(p)* of the ellipsoid's height to its depth is fixed,

$$\text{where} \quad p = \frac{radius_y}{radius_z} \qquad (5)$$

Then the length of the ellipsoid *(L)* is given by $L = (2 * radius_x)$, and the new dimensions are determined as follows:

$$radius_x{}' = L/2 \qquad (6)$$

$$radius_y{}' = radius_z{}' * p \qquad (7)$$

$$radius_z{}' = (3v/4\pi\, radius_x{}'\, p)^{1/2} \qquad (8)$$

A valid 2D muscle deformation is illustrated in Figure 5. As the leg stretches (Figure 5a and Figure 5c) the muscle elongates while the width of the muscle decreases. During the muscle flexion the muscle's length shrinks and the width increases.

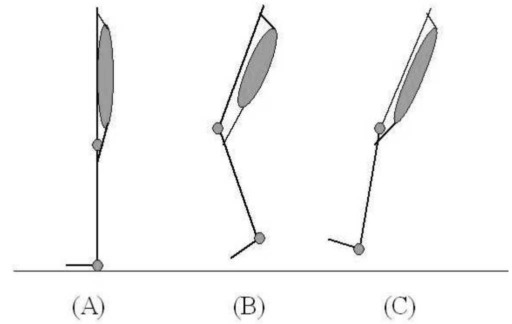

Figure 5: *Muscle Shape Transition During Motion*

4.2 Surface Layer LOD

The main motivation for the use of super-ellipsoids within the ALOHA framework is their simple visualisation as well as their potential for faster tessellation. The tessellation itself can be controlled by altering the step-size of the *theta* and *phi* parameters in equations (1-3). The number of springs used to model the fatty tissue layer will be directly proportional to the number of vertices in the muscle's primitive. Furthermore, since the skin mesh is comprised of the outer spring-ends in the fat layer, the geometric level of detail of the skin is also directly determined from the muscle tessellation. This simple technique allows the LOD resolver to decide the complexity of a given super-ellipsoid's appearance using a single parameter.

5. Animation

An important requirement in the ALOHA framework is that the characters have the ability to interact with dynamically changing environments. Objects that the character may have to interact with will most likely not have predefined positions, so actions such as reaching and touching will have to be computed at runtime for believability to be preserved. Inverse kinematics is a useful tool in such situations. It can solve for the joint angles of an articulated model given the positions of the root and end-effector and the lengths of the links.

There are numerous ways to implement inverse-kinematics, each with their own respective advantages and disadvantages.

The inverse kinematics technique adopted for testing on the ALOHA framework is based on cyclic coordinate descent (CCD), an algorithm from the field of robotics [12]. It is an iterative heuristic search technique based on non-linear programming techniques and forward recursion formulae.

The CCD method works by considering one joint at a time, from the tip to the base of the articulation.

Given a current end-effector position,

$$P_c = (x_c, y_c, z_c), \qquad (9)$$

and a current end-effector orientation specified as

$$O_c = [u_{1c}, u_{2c}, u_{3c}]^T, \qquad (10)$$

we can find a joint vector q which minimises the error measure

$$E(q) = E_p(q) + E_o(q), \qquad (11)$$

in order to move the end-effector as close as possible to a desired position P_d and orientation O_d.

The CCD method is particularly suitable because it is numerically stable, is immune to singular configuration problems and can be applied to serial manipulators having any number of degrees of freedom. Furthermore, it is computationally efficient, which is especially important when considering a framework that may have to maintain real-time performance for a large number of virtual characters.

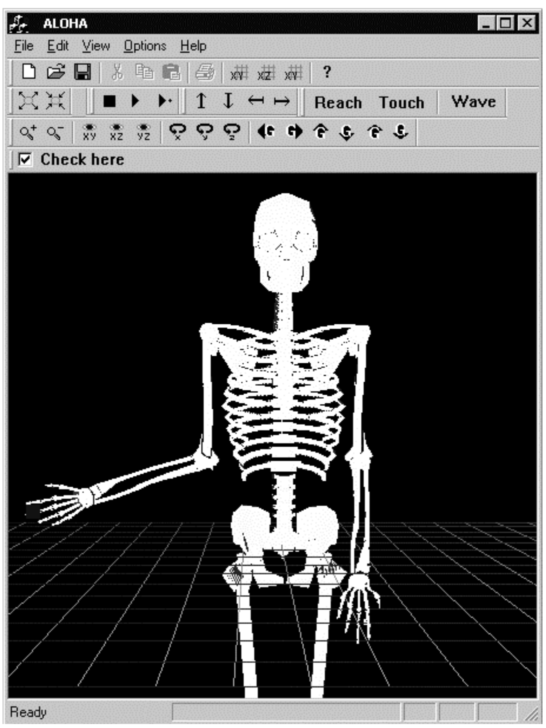

Figure 6: *The ALOHA inverse kinematics test-bed (skeleton model courtesy of 3D Cafe www.3dcafe.com).*

5.1 Animation LOD

In order to provide a scalable framework, different animation levels of detail may be requested by the LOD resolver. For certain tasks, the inverse kinematics will be allocated less iteration's with which to resolve the joint angles. Furthermore, movements rated with a low importance by the LOD resolver may receive fewer frames of actual animation. Although such techniques will sacrifice simulation accuracy and smoothness for speed, the LOD resolver should only approve of such measures in cases where the viewer will not perceive them.

Despite the computational efficiency of the CCD technique, it does suffer from poor convergence rates, particularly where high accuracy is required. Because of this drawback, another inverse kinematics technique with a better convergence rate will be employed where greater accuracy is required. Since greater accuracy also implies greater computational complexity, this technique will be reserved for animations that are highly rated by the LOD resolver.

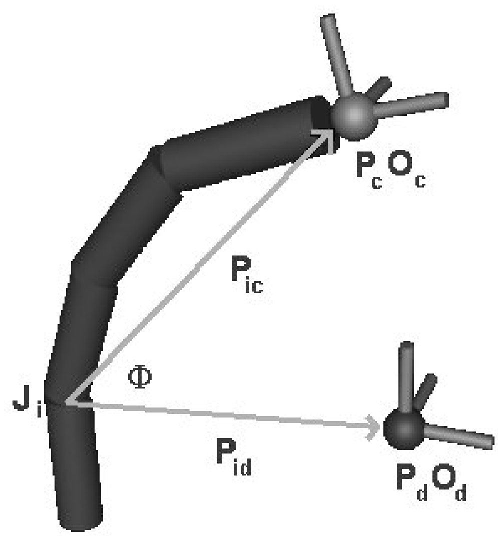

Figure 7: *A CCD iteration step for the joint i. The end-effector is marked as a blue sphere (top sphere) and the desired end position is marked as a red sphere (bottom sphere).*

On the other end of the level of detail scale are tasks that require particularly low detail. For these low importance tasks, predefined forward kinematics from motion capture data may be used to obtain high-speed, low-detail animation.

The ALOHA framework also caters for cyclic motion such as walking on a number of levels. Depending on the required level of detail, as directed by the LOD resolver, the system may, either use a purely kinematic or dynamic method.

In the simplest case, the system uses a simple kinematic interpolation technique to produce the required cyclic motion. If a high level of realism is required, the system uses dynamic techniques to simulate the motion in question. We use a dynamic constraint based technique, such as the one described by Barr & Barzel [2] for such simulations. For further control over our simulation during high level of detail requirements, a space-time constraints method such as the one described by Witkin & Kass [15] or Gortler et al [6] could have been used instead but we leave that for future work.

A lower level motion capture technique is also used to supplement the required realism of our motion whenever the system deems that that level of detail to be applied is sufficiently low enough.

6. Conclusions and Future Work

We have presented a LOD framework that is capable of selecting and processing both geometric and animation detail levels for virtual humans. The framework remains flexible to deal with different LOD resolution criteria, such as distance to viewer, complexity of the environment and hardware installed. The scalability of the framework provides support for a number of multi-layered virtual characters while retaining real-time frame-rates.

Presently the ALOHA framework advocates the use of a 4-layered character model. However, it is not restricted to this model, since theoretically the LOD resolver is free to dictate any number of geometric parameters to optimise performance, while maintaining an acceptable appearance for each object. Future geometric extensions will involve the implementation of techniques used by Turner & Gobbetti [11] to enhance the physical behaviour of the skins surface.

An important area of future research involves switching between animations. This is important not only for blending different animations at the same level of detail, but also for smoothly switching between similar motions that are using different levels of detail. This could be viewed as the animation equivalent of geo-morphing, a technique that is used to smooth level of detail changes for geometries.

Another route open to interesting future research is that of a multi-resolution layering model. Successive layers could be ‘peeled’ off less important characters in order to provide simpler and less computationally expensive models. However, it is likely that any such system would need to support a large number of different animation and geometric techniques in order to prove fruitful.

Acknowledgements

This work is sponsored by Enterprise Ireland strategic research grant EI ST/99/086. Finally thanks to James Sinnot for his help with super-ellipsoids.

References

1. Barr A.H, *“Superquadrics and angle preserving transformations”*, IEEE Transactions on Computer Graphics and Applications, 1(1), 11-23, January 1981.

2. Barr A.H, Barzel R, *“A Modelling System Based On Dynamic Constraints”* Proc. SIGGRAPH 1988, pp. 179-188.

3. Carlson D.A, Hodgins J.K, *“Simulation levels of detail for real-time animation”*, In Proc. of Graphics Interface ’97, pages 1-8, 1997.

4. Chadwick J.E, Haumann D.R, Parent R.E, *“Layered Construction for Deformable Animated Characters”*, Proceedings of SIGGRAPH 89, Computer Graphics Proceedings, 23 (3), pp. 243-252 . Edited by Jeffrey Lane.

5. Cozot R , Multon F, Valton B, Arnaldi B, *“Animation Levels of Detail Design for Real-Time Virtual Human”*, Eurographics Workshop on Animation and Simulation’99, 1999.

6. Gortler S.J, Liu Z, Cohen M.F *“Hierarchical Spacetime Control”* Proc. SIGGRAPH 1994, pp.35-42.

7. Heckbert P. and Garland M. *“Multiresolution modeling for fast rendering.”* In Proc. of Graphics Interface ’94, pages 43-50, 1994.

8. Scheepers F, Parent R.E, Carlson W.E, May S.F,*“ Anatomy-Based Modeling of the Human Musculature ”*, Proceedings of SIGGRAPH 97, Computer Graphics Proceedings, Annual Conference Series, pp. 163-172 . Addison Wesley. Edited by Turner Whitted. ISBN 0-89791-896-7.

9. Singla A, Ramachandran U, Hodgins JK, *“Temporal notions of synchronization and consistency in beehive”*, In Proc. of the 9th Annual ACM Symposium on Parallel Algorithms and Architectures, 1997.

10. Sinnott J, Howard T, *“SQUIDS: Interactive Deformation of Superquadrics for Model Matching in Virtual Environments”*, Eurographics UK Conference Proceedings 2000, pp. 73-80.

11. Turner R, Gobbetti E, *“Interactive Construction and Animation of Layered Elastically Deformable Characters”*, Computer Graphics Forum, 17(2), pp. 135-152 (1998). Blackwell Publishers. ISSN 1067-7055.

12. Wang L.C.T , Chen C.C, *"A Combined Optimization Method for Solving the Inverse Kinematics problem of Mechanical Manipulators"*, IEEE Transactions on Robotics and Automation, 7(4):489-499, August 1991.

13. Welman C, "*Inverse Kinematics and Geometric Constraints for Articulated Figure Manipulation"*, Masters Thesis, 1993.

14. Wilhelms J, *"Animals with Anatomy"*, IEEE Computer Graphics & Applications, 17 (3), pp. 22-30 (May - June 1997). ISSN 0272-1716

15. Witkin A, Kass M, *"Spacetime Constraints"* Proc. SIGGRAPH 1988, pp. 159-168.