

# Hair Motion Simulation

Yusuf Sahilliođlu and Bülent Özgüç

Bilkent University, Department of Computer Engineering  
Bilkent, 06800 Ankara, Turkey  
sahilli@ug.bilkent.edu.tr, ozguc@bilkent.edu.tr

**Abstract.** Hair motion simulation in computer graphics has been an attraction for many researchers. The application we have developed has been inspired by the related previous work as well as our own efforts in finding useful algorithms to handle this problem. The work we present uses a set of representations, including hair strands, clusters and strips, that are derived from the same underlying base skeleton, where this skeleton is animated by physical, i.e. spring, forces.

## 1 Introduction

One of the most exciting and challenging research areas in CG is the hair motion simulation. The challenge is due to the fact that a human head has up to 160,000 individual hair strands and simulating each of them in a straightforward manner is quite costly and tedious. Therefore, one should come up with efficient algorithms.

One idea is using level of detail (LOD) approach, a popular method in CG modeling. The hair can be represented as segments where each segment might be either a hair strand, a cluster or a strip. Depending on the distance between the viewer and animated head, the corresponding level switches will be performed.

In addition to the LOD usage, there is also another useful idea: The structure behind the modeling of these segments. All the segments are derived from the same base hair skeleton structure and the dynamics of this structure is based on the applied physical forces, such as spring, gravitation, friction, absorption and repulsion force.

This paper aims to present the work we have carried out to add these features to our hair motion simulation application. We should also produce a human head which is ready to host the hair. Thus, this problem can be described as the modeling, dynamics and rendering of human hair and head in real time.

## 2 Solution and Design Decisions

The solution to this problem consists of two parts, namely, modeling, rendering and animating human head, and modeling, rendering and animating human hair. While deciding on the design issues, a human head model for hair placement is prepared, making use of an auxiliary software package called 3DS Max [1]. Then, use of the physical laws and the base skeleton structure for hair is prepared. Finally, the hair part is implanted on the scalp of the head part.

## 2.1 Human Head

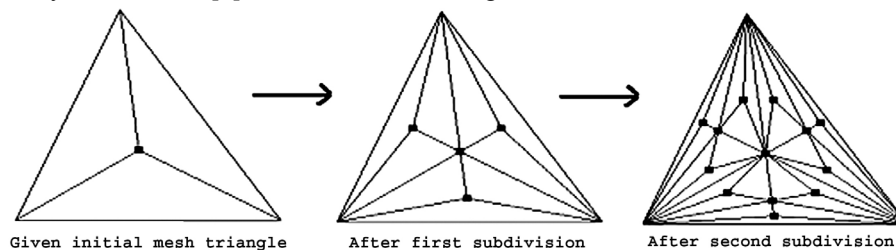
The procedure to obtain a human head requires modeling and rendering it. Making use of an existing head model in 3DS Max software environment, the head is loaded into the software and some transformations and a crucial operation, scalp mesh separation are added to it. Having separated the scalp mesh and added some basic transformations, i.e. rotation and translation to the head, the hair can be implanted and moved.

**Tasks of Scalp Mesh Triangles.** Each scalp mesh triangle is associated with the following tasks:

1. Determining the implanting point of the hair segment (rooting)
2. Determining the density of hair (density)
3. Detecting collision between scalp and hair segment (collision)

*Rooting:* The root of each hair segment will be the center of gravity (COG) of the corresponding scalp mesh triangle [2]. Having obtained the scalp mesh triangles, new hair segments originating from the COG of these triangles are generated. Since each hair segment will be long enough, the segments originating from a higher triangle, will also cover regions of some lower triangles. As the number of segments increases, some of the lower level triangles will be automatically hidden by the long hairs originating from above. Hence, for the sake of performing less computation in real-time, the rooting of hair from lower triangles is sometimes omitted.

*Density:* If the number of hair segments is equal to the number of scalp mesh triangles, rooted hair might be sparse. In order to handle this problem, a given triangle is recursively subdivided and new hair segments are originated from the COG of the newly created ones [2], as demonstrated in Figure 1.



**Fig. 1.** Increasing number of roots via subdivision process

It should be noted that when the type of the triangle is close to an equal-sided triangle (well-proportioned), the rooting points are fairly and logically distributed over the shape. However, when the triangle sides are badly proportioned then the rooting points will stack on one region of the shape dominantly. The latter case is undesirable and therefore while generating the scalp mesh triangles, it should be tried to create as proportioned triangles as possible.

*Collision:* The last task of the scalp mesh triangles is the collision detection between themselves and the corresponding hair segment. To do this the normal -to decide collision- and some physical constants (i.e. absorption, friction, repulsion) -to decide the response forces in case of a collision- of each triangle are necessary. Since each

hair segment is recognized by their masses<sup>1</sup>, the location of all masses is compared with a single scalp triangle. In case of a collision between a given triangle and a hair segment mass point, the following collision response operations are performed:

1. Copy and omit the velocity component of the mass in each axis, set  $v_x$   $v_y$   $v_z$  to 0, in order to prevent it from going further towards to the inside of head.
2. Apply the friction force, using the physical friction constant of triangle material, to the mass. The force will be proportional to the magnitude of velocity (which is copied before setting initial velocity to 0) of the collided mass and be applied in the direction of the velocity vector of the mass to reduce repulsion effect.
3. Apply the absorption force, using the physical absorption constant of triangle material, to the mass. The force will be proportional to the magnitude of velocity of the collided mass and be applied in the direction of the velocity vector of the mass to reduce repulsion effect.
4. Apply the repulsion force, using the physical repulsion constant of triangle material, to the mass. The force will be proportional to the magnitude of velocity of the collided mass and be applied in the *opposite* direction of the velocity vector of the collided mass to make mass go away from triangle.

Considering that there are roughly 80 mass points for one hair segment and each of these points must be compared with candidate triangles for collision detection process, the cost of computation becomes expensive if the number of triangles in scalp increases. Therefore, some sparseness in the scalp is left and those sections are treated as rough bounding rectangles.

## 2.2 Human Hair

There are three main aspects in human hair simulation which are stated as hair shape modeling, hair dynamics (animation) and hair rendering in [3].

Hair modeling involves the exact or fake creation of individual hairs. Since there are approximately 160.000 individual hair strands on a given human scalp, the modeling should be performed in such a way that the resulting animation of the selected model will be efficient and fast.

Hair dynamics involves the animation of the hair. Several forces are applied to the particular parts of the hair in order to accomplish the animation.

Hair rendering is involved with the appearance of simulated hair from the viewer's point of view. Several rendering techniques are applied to the hair for realism.

**Hair Modeling.** There are three hair modeling schemes dominant in this research area: Strand Hair Model (SHM) introduced in [4, 5]; Multiresolution Hair Model (MHM) introduced in [6]; Modeling with Level-of-Detail Representations (LOD) introduced in [7–9].

*Strand Hair Model (SHM):* SHM is the simplest way to model the hair. Each individual hair strand is explicitly designed by this scheme [5]. The details can be seen in Figure 2.

---

<sup>1</sup> Masses are the nodes of the base skeleton structure.

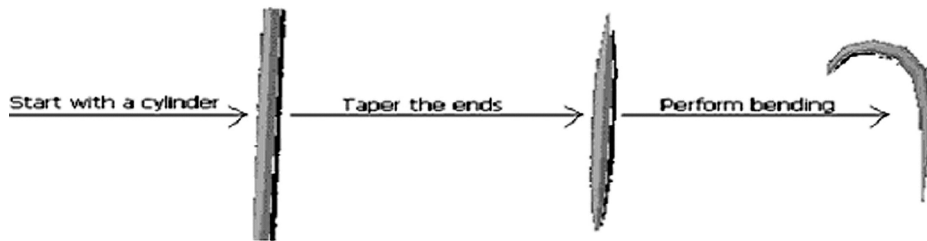


Fig. 2. The process of modeling an individual hair strand in SHM

After applying the related transformations, namely tapering and bending respectively to the initial cylinder object, an individual hair strand is obtained.

*Multiresolution Hair Model (MHM)*: MHM is an admissible model to represent hair since it is based on the multiresolution concept whose major benefit is the user's freedom to choose the appropriate level of detail for a desired model manipulation. Multiresolution manipulations for hair modeling are achieved with a hierarchy of a set of generalized cylinders. Hair design is created with a small set of clusters, roughly 30 clusters per human head. Subdividing these clusters yields the submodel of MHM, which is Cluster Hair Model (CHM). Putting all together, MHM combines the benefits of CHM and SHM by allowing local control as well as global control. MHM is too complex to model since it allows for interactive hair modeling. Considering that our project does not deal with interactively creating complex hair models, we have concentrated on the following method instead of using the MHM.

*Modeling with LOD Representations*: Getting the basic idea from [7], we introduced some new algorithms. The preferred approach, LOD, uses three novel representations based on a base skeleton to create levels-of-detail (LODs) for hair. These are strips, clusters and individual strands:

*Strips*: Strip is the coarsest (lowest) LOD used for hair modeling. It is typically used to represent the inner most layers of hair and it is responsible for the global physical behavior and the volume of the hair during the simulation. The transition (switching) to this LOD representation is performed when the viewing distance to the human head is increased.

In order to model the strip, the base skeleton structure is used. Starting from the top skeleton node, it is proceeded towards to the bottom by putting 2 control points for each of the node point encountered. It is important to notice that, the 2 control points and the related node point are collinear. Obtaining the global surface by this way, we subdivide it to obtain a smooth representation. The subdivision surface that is obtained for one strip is shown in Figure 3.

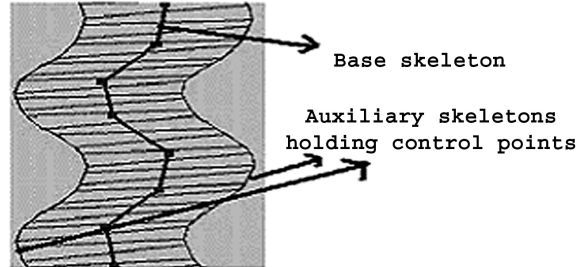


Fig. 3. Strip represented as a subdivision surface

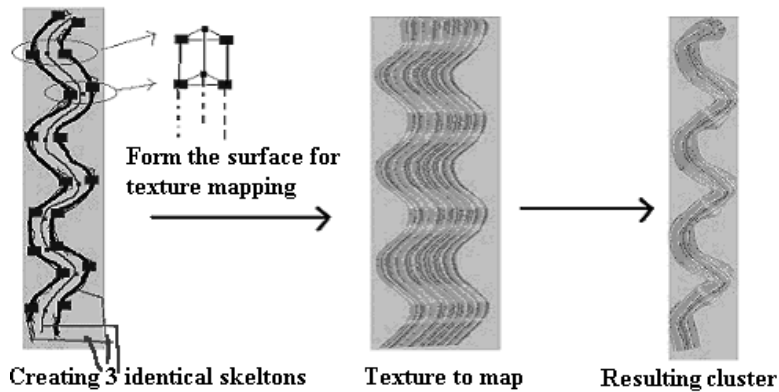


Fig. 4. Process of obtaining a cluster

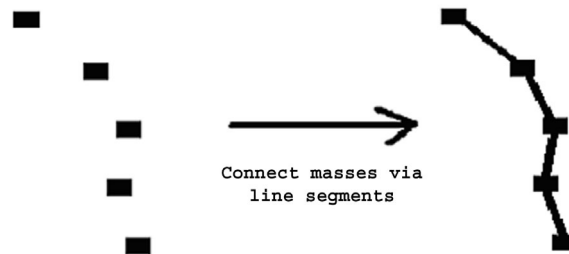
*Clusters:* Cluster is used to represent a group of hair. This LOD improves performance since instead of animating lots of hair strands close to each other; we will animate just one object that represents the total behavior.

In addition to the use of base skeleton structure, the generalized cylinder concept arises in order to model a cluster. The generalized cylinder is obtained as follows: Initially three identical base skeletons are created; one on the left, one on the right and the other on the back (Figure 4). The corresponding nodes of each skeleton are joined, forming a surface to which our texture can be easily mapped. The texture gives the illusion of a group of strands, roughly 30 strands. Hence, by applying forces to just 3 skeletons, a group motion consisting of 30 strands is obtained. It will gain a volume to hair and reduce the sparseness with the cost of animating just three skeletons.

*Individual Strands:* An individual strand is used to model the hair strand and can be seen only when the viewer is very close to the head. Switching to this LOD will take place when distance between the viewer and the head is sufficiently decreased.

As in all other LODs, individual strands are based on the base skeleton. Actually, the base skeleton itself is directly the hair strand. By combining two consequent

masses of the skeleton via a line segment, the hair strand is automatically obtained (Figure 5).



**Fig. 5.** Process of obtaining an individual strand

**Hair Dynamics.** Hair dynamics (animation) is closely related to the base skeleton structure. The base skeleton is defined as the structure consisting of point-like masses connected to each other via line segments (Figure 5). The hair dynamics is performed by applying forces to each mass of the base skeleton structure, where the term mass seems logical since the Newton forces ( $F = ma$ ) is applied to those masses. The applied forces will change the position of each mass accordingly. Changing the positions of masses will change the position of the base skeleton and therefore the position of the active LOD representation derived from the moved base skeleton.

*Moving Base Skeleton.* Moving the base skeleton corresponds to moving the derived LOD representation as well. Thus, the understanding of the idea of moving this skeleton resolves the dynamics involved.

The source of a movement is the applied forces. The net force applied on a mass,  $F$ , changes its acceleration, by  $F = ma$  where  $m$  is predefined. By new acceleration, the new velocity is obtained and by new velocity the displacement that should be added to the current position for that particular mass is obtained.

There are two types of forces (main forces) that are always applied regardless of the animation state. *Spring forces:* Two consequent masses introduce to each other spring forces. *Gravitation forces:* The traditional gravitational force is applied to each mass.

Besides, there are four more forces (extra forces) that are applied when necessary:

1. Air Friction: As long as a mass is moving (velocity is not 0), an air friction force is applied in the opposite direction of velocity vector and with the magnitude proportional to velocity of the mass.
2. Scalp Friction: As long as a mass is colliding to a triangle, or a bounding rectangle of scalp mesh, a friction force is applied in the direction of velocity vector and with the magnitude proportional to velocity of the mass.
3. Scalp Absorption: As long as a mass is colliding to a triangle, or a bounding rectangle of scalp mesh, an absorption force is applied in the direction of velocity vector and with the magnitude proportional to velocity of the mass.

4. Scalp Repulsion: As long as a mass is colliding to a triangle, or a bounding rectangle of scalp mesh, a repulsion force is applied in the *opposite* direction of velocity vector and with the magnitude proportional to velocity of the mass.

The visualization of the forces mentioned above is given in Figure 6.

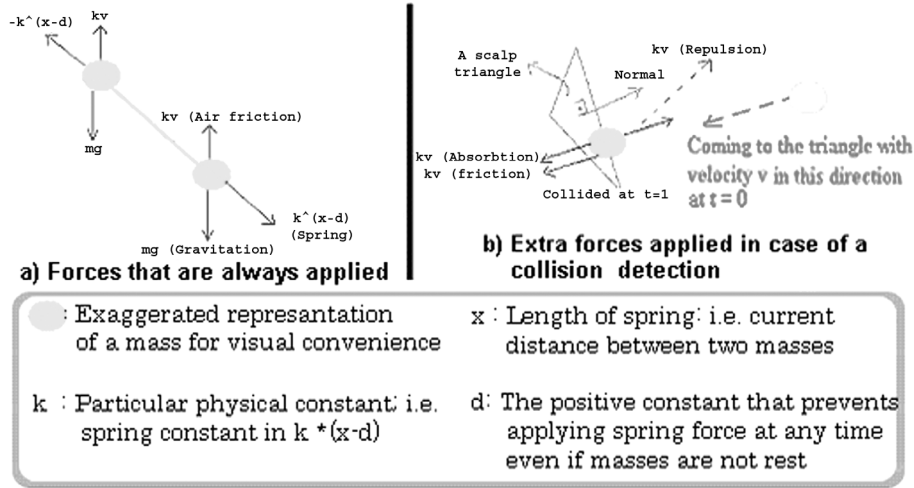


Fig. 6. Visualized representation of forces

In each frame of the animation, the current total force acting on corresponding mass is simulated by using the Newton law,  $F = ma$ . Since  $F$  is known, by the result of Figure 6, and  $m$ , the predefined constant is known, acceleration,  $a$ , is computed. Knowing the acceleration, the amount of change in velocity is computed by using  $v = ad_t$  where  $a$  is the acceleration just found and  $d_t$  is the predefined change in time. Adding this new  $v$ , the velocity gained/lost, to current  $v$ , new velocity is obtained.

Knowing the new velocity, the amount of change in displacement is computed by using equation  $x = vd_t$ , where  $v$  is the velocity just found and  $d_t$  is the change in time. Adding this new  $x$ , the road taken, to current  $x$ , the desired new position is obtained.

The scalp friction-absorption-repulsion forces need to detect collision. Another collision detection is necessary between the hair segments themselves. All of these collisions will change the net force when they are detected and the changing net force will obviously change the dynamics (movement) of the hair.

*Collision Detection.* There are two types of collisions to be detected. When they are detected, the specified response forces of Figure 6 are applied to the collided mass.

In *scalp collision*, it is required to detect collision between a scalp mesh triangle and a mass of the hair segment. This is done in two steps. First, the mass must reside on the “infinite” plane of the candidate triangle. If not, then there is no need to apply the second step since the mass cannot hit the triangle. However, if the mass is on the plane then the second step takes place to decide whether the mass is inside the boundaries of the candidate triangle. To perform this inside-the-boundary test, we

create three planes for each edge of the candidate triangle with their normals all pointing in. The mass is decided to hit the triangle if and only if it is inside all of the three planes. Applying these two steps to all of the mesh triangles reduces the real-time running performance significantly. Therefore, *binary search* approach is used to find the candidate triangle. This is done by a bounding box approach where the far left triangles are ignored if the mass is on the right side with respect to the center of gravity of current region of head and vice versa (Figure 7).

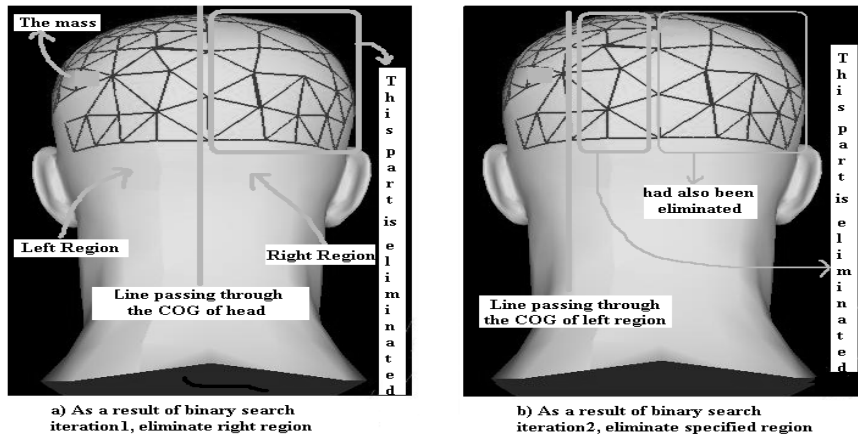


Fig. 7. Eliminating irrelevant triangles before deciding the one being hit by the mass

In *hair collision*, the simplest algorithm is the detection of collision of every mass of each hair strand with all the masses of the remaining hair segments exhaustively. This is done by comparing the nearness of the two mass coordinates; i.e. by tolerating some amount of error. This tolerance is necessary since it is almost impossible to match two coordinates exactly when many forces are acting on each mass in each refresh cycle.

This approach is computationally expensive. For only one mass of  $N$  masses of the given segment, all  $N$  masses of the other hair strands are tested, leading to the cost of  $O(N^2)$  for just one hair collision. If we assume that there are  $N$  such segments on the scalp, this cost increases to  $O(N^3)$ . In short, we will be forced to apply this  $O(N^3)$  algorithm in each frame, which is done in periods of 0.33 seconds for 15 strands with 80 masses each. The bounding box idea can help. Instead of testing the given mass with all of the  $N$  masses of the candidate hair segment all the time, we put that candidate segment into a bounding box and make a test only once, just for the intersection between mass and box. Only if this initial test is passed, i.e. intersection is detected, then we try the test of one mass with all others. The boxing of a given hair segment will be done using the min/max of axis values as Figure 8 illustrates.



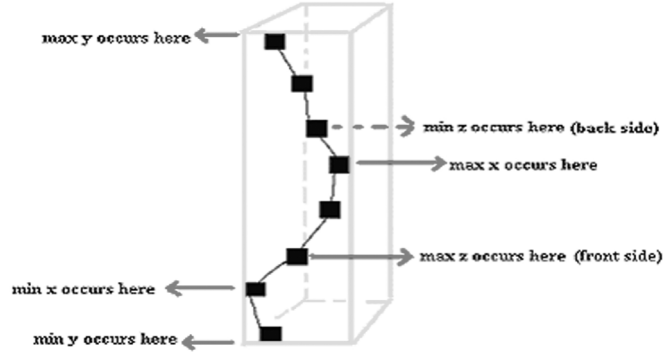


Fig. 8. Bounding box of a given hair segment

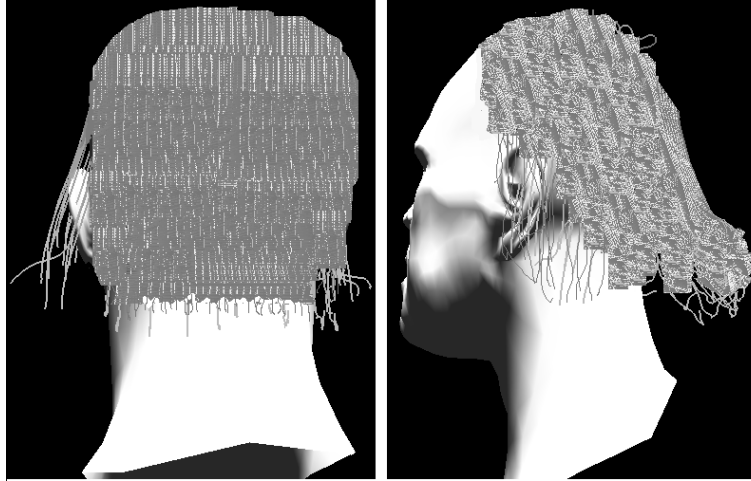


Fig. 9. Rendered versions of cluster representation

**Hair Rendering.** According to the active LOD, several techniques of rendering are applied. When the viewing distance is very large, we switch to strip representation. Since strip is a subdivision surface, it has well-defined surface normals. Hence, bump texture maps are easily applied. Clusters are also bump-mapped since they have similar properties (Figure 9). Strands are just lines, thus they are drawn with a single shaded color.

### 3 Results and Future Work

Hair motion simulator is tested on a PC having 512MB of RAM and 2.4GHz processor speed. In order to evaluate the program several tests, whose results are given in Table 1, have been applied. (SC: Scalp to hair collision; HC: hair to hair collision)

**Table 1.** Frame process time measured in seconds

	no collision	SC	HC	SC + HC
1 Hair Strand	0.04	0.04	n/a	0.04
100 Hair Strands	0.08	0.27	0.88	>2.0
1 Hair Cluster	0.05	0.05	n/a	0.05
100 Hair Clusters	0.09	0.44	0.94	>3.0

The model used for the table consists of 17064 vertices for the head mesh that make up 5688 head mesh triangles, 312 scalp mesh vertices for 104 scalp mesh triangles and 80 masses for each hair strand. Real time animation for a denser hair placement than what is given in Table 1 is not achieved yet and we are improving some of the techniques used so that a more realistic head (Figure 9) can be animated close to real time.

## References

1. Ben, D.: Game programming with a personality. <http://gametutorials.com>
2. Usher, J.: Computer Science Honours Dissertation 499: Dynamic Hair Generation. Curtin University of Technology, <http://www.vhml.org/theses/usher/>
3. Thalmann, N., Hadap, S., Kalra, P.: State of the art in hair simulation. In: Int. Workshop on Human Modeling and Animation, Seoul, Korea. (2002) 3–9
4. Anjyo, K., Usami Y., Kurihara, T.: A simple method for extracting the natural beauty of hair. In: Proc. SIGGRAPH'92. (1992) 111–120
5. Tomwool, H.: Growing hair by tutor Tomwool. [http://luna.spaceports.com/~inferno/tu\\_hair.htm](http://luna.spaceports.com/~inferno/tu_hair.htm)
6. Kim, T., Neumann, U.: Interactive multiresolution hair modeling and editing. ACM Transactions on Graphics **21**(3) (2002) 620–629
7. Ward, K., Lin, M., Lee, J., Fisher, S., Macri, D.: Modeling hair using level-of-detail representations. In: Proc. of Computer Animation and Social Agents, New Brunswick, New Jersey. (2003) 41–47
8. Ward, K., Lin, M.: Adaptive grouping and subdivision for simulating hair dynamics. In: Proc. of 11<sup>th</sup> Pacific Conference on Computer Graphics and Applications, Canmore, Canada. (2003) 234–243
9. Ward, K., Fisher, S., Lin, M.: Simplified representations for modeling hair. UNC Technical Report #TR02-020 (2002)