A Simulation-based VR System for Interactive Hairstyling*

Kelly Ward† Nico Galoppo Ming C. Lin

The University of North Carolina at Chapel Hill
{wardk,nico,lin}@cs.unc.edu

ABSTRACT

We have developed a physically-based VR system that enables users to interactively style dynamic virtual hair by using multi-resolution simulation techniques and graphics hardware rendering acceleration for simulating and rendering hair in real time. With a 3D haptic interface, users can directly manipulate and position hair strands, as well as employ real-world styling applications (cutting, blow-drying, etc.) to create hairstyles more intuitively than previous techniques.

1 INTRODUCTION

Virtual environments created for interactive hairstyling can be used to understand and specify detailed hair properties for several applications, including cosmetic prototyping, education and entertainment, and cosmetologist training. Accurate virtual hairstyling requires both high performance simulation and realistic rendering to enable interactive manipulation and incorporation of fine details. The appearance of hairstyles results from physical properties of hair and hair mutual interactions. Therefore, hair dynamics should be incorporated to mimic the process of real-world hairstyle creation. However, due to the performance requirement, many interactive hair modeling algorithms tend to lack important, complex features of hair, including hair interactions, dynamic clustering of hair strands, and intricate self-shadowing effects.

An intuitive virtual hairstyling tool needs to take into account user interaction with dynamic hair. Until recently, the complexity of animating and rendering hair had been too computationally costly to accurately model hair's essential features at desired rates. As a result, many hairstyling methods ignore dynamic simulation and/or user interaction, which creates an unnatural styling process in comparison to what would be expected in practice.

Main Results: In this paper, we present a *physically-based* virtual reality system that mimics real-world hairstyling processes and requires no knowledge other than common hair manipulation techniques. By using multi-resolution simulation techniques and programmable graphics hardware, we developed a physically-based virtual hair salon system that animates and renders hair at accelerated rates, allowing users to interactively style virtual hair in a natural manner. With an intuitive 3D haptic interface, users can directly manipulate and position hair strands, as well as employ real-world styling applications (e.g. cutting, wetting, applying styling products) to create hairstyles as they would in the physical world. The main characteristics of our system are the following:

Figure 1: Hairstyle interactively created using our system.

- **Direct 3D hair manipulation with a haptic interface**: We use a commercially available haptic device to provide an intuitive 3D user interface, allowing the users to directly manipulate the hair in a manner similar to real-world hairstyling.
- **Visually plausible rendering**: By exploiting the capability of programmable graphics hardware and multi-resolution representations, we can render plausible hair appearance due to self-shadowing, wet surfaces, etc. in real time on current commodity desktop PCs.
- **Multi-resolution hairstyling**: We achieve interactive hair simulation by using level-of-detail representations, which accelerate dynamics computation and enable adaptive hair clustering and subdivision on the fly.
- **Physically-based interaction**: By modeling hair's properties and dynamic behavior in the presence of water and styling products, we introduce the ability to interactively apply hairspray, wet, blow-dry, cut, and manipulate hair as in the physical world, like no other systems can at present.

Figure 1 illustrates a hairstyle created by a naive user using our virtual hairstyling system in less than 10 minutes.

Organization: The rest of this sketch is organized as follows. Related work is briefly reviewed in Section 2. Section 3 presents the user interface for the system. The dynamic simulation and rendering of hair are described in Section 4. Details of user interaction and application features are discussed in Section 5. Finally, we conclude with some results and possible future research directions in Section 6.

2 RELATED WORK

Hair modeling involves hair shape modeling, dynamic hair simulation, and hair rendering. An overview of work in these areas can be found in [5]. Our summary of related material is limited to the multi-resolution simulation and rendering methods this work extends.

---

*Supported in part by Army Research Office, National Science Foundation, Office of Naval Research, and Intel Corporation.

†now at Walt Disney Feature Animation

We use the three LOD representations and adaptive grouping and splitting process introduced in [11, 10]. The representation and resolution of a volume of hair is controlled by a hair hierarchy that is constructed by the continual subdivision of hair strips, clusters, and strand groups. Throughout the simulation, the hair hierarchy is traversed on the fly to find the appropriate representation and resolution for a given section of hair adaptively.

In [11, 10], an appropriate LOD representation was chosen based on the hair's importance to the application. Sections of hair that could be viewed well were simulated and rendered with high detail. Also, as a section of hair moves rapidly, a high LOD was used to capture the intricate detail, similarly performed by [1]. These criteria aided in accelerating simulation and rendering without losing much visual fidelity.

In this work, we couple the hair hierarchy with a simulation localization scheme (see Section 4.2) to achieve interactive hair animation. Moreover, our LOD selection criteria differ from that of [11, 10] to include areas of user interaction and an additional emphasis on the hair's motion.

3 USER INTERFACE

Our prototype system uses a SensAble Technologies' PHANToM as a 3D user input device. The real-time display of the PHANToM input is rendered using a commercial haptic toolkit called *GHOST*. The position and orientation of the device are updated and rendered at each frame.

A 2D menu is projected onto the 3D scene containing the avatar and hair model. Figure 2 illustrates a user operating the system. The user can interact with both the 3D scene and 2D menu using the PHANToM stylus in a seamless fashion. The user interactively positions the stylus over the 2D menu icons and pushes the stylus button to choose the desired application. The position, orientation, and range of influence of the application are depicted in the scene. As the user moves the stylus, the application's area of influence interactively follows the position and orientation of the user's hand in 3D. The camera is controlled through the mouse.

Figure 2: User Interface: PHANToM provides 3D user input and 2D menu buttons are labeled with icons to show applications.

4 INTERACTIVE HAIR SIMULATION AND RENDERING

Hairstyling in the natural world is performed by focusing on specific sections of hair and executing a desired task to a given area. This focus correlates naturally with the use of multi-resolution techniques to simulate hair. Moreover, accelerated hair rendering is required for user interaction.

4.1 Dynamics and Collision Detection

Each LOD hair representation (strips, clusters, and strands) follows the same dynamics model for motion. We use the dual-skeleton system introduced by [9] for controlling the hair's dynamics. The dual-skeleton system is compelling in that it can capture the details of typical dry hair as well as wet hair and hair with hairspray, or some other styling products applied.

We have utilized the localized collision detection model of [9] that is based on the dual-skeleton setup and the family of *swept sphere volumes* (SSVs). A swept sphere volume is created by taking a core shape, such as a point, a line, or a rectangle, and growing outward by some offset. The SSVs encapsulate the hair geometry (of any type or resolution LOD) and are used as bounding volumes for collision detection.

Both hair-hair and hair-object collision detection is performed by checking for intersection between corresponding pairs of SSVs; this is done by calculating the distance between the core shapes and subtracting the appropriate offsets. Hair-object collisions are handled by moving the hair section outside of the body and that hair section is restricted to only move tangential to or away from the object, based on the object's normal direction. Hair-hair collisions are processed by pushing the hair sections apart based on their respective orientations as explained in methods by [8, 10].

4.2 Simulation Localization

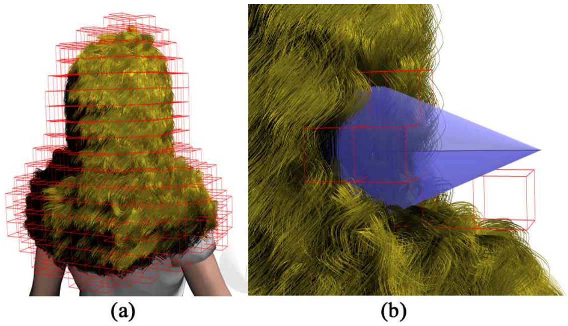

Figure 3: (a) Shows all of the grid cells that contain hair geometry (b) Highlights the cells that will be effected by the current application (applying water).

We use a spatial decomposition scheme to rapidly determine the high activity areas of the hair; these areas are then simulated with finer detail. We use a uniform grid consisting of axis-aligned cells that encompass the area around the hair and human avatar. Spatial decomposition schemes have been utilized previously for hair-hair interaction methods where sections of hair that are located in the same cell will be tested against each other for overlap. We have extended this process to all features of hair simulation, not just collision detection.

4.2.1 Insertion into the Grid

The polygons of the avatar, or other objects, are placed into the grid to determine potential collisions with the hair. Object positions only need to be updated within the grid if the object is moving, otherwise the initial insertion is sufficient.

Every time a section of hair moves, or the skeleton for simulation is updated, its line swept spheres (LSSs) or rectangular swept spheres (RSSs) are inserted into the grid. An SSV is inserted into the grid by determining which cells first contain the core shape of the SSV (line or rectangle), then the offset of the SSVs are used to determine the remaining inhabited cells. Figure 3(a) shows the grid cells that contain hair geometry.

When the user employs an application (e.g. spraying water, grabbing the hair) the grid is used to indicate which portions of the hair are potentially affected by the user's action. As the user moves the PHANToM stylus, its position and orientation are updated. Each application has an *area of influence* that defines where in space its action will have an effect. This area is defined as a triangle for the cutting tool and a cone for the remaining tools. The cone of influence is defined by the application's position, orientation, length, and cutoff angle. These properties define the cone's position in the grid. Inserting the cone becomes similar to inserting an LSS, but the offset becomes a variable of radius along the core line (an SSV has a constant offset along its core shape). The triangle for cutting is defined by the space between the open blades of a pair of scissors.

4.2.2 Retrieval from the Grid

Grid-cells that contain both impenetrable triangles (from the avatar or another object in the scene) and hair geometry are marked to be checked for hair-object collision. Only these cells contain a potentially colliding pair. Similarly, any grid cell containing more than one section of hair is marked to be checked for hair-hair collisions.

Likewise, the grid maintains a list of grid-cells where the user interaction cone or triangle has been inserted. Any of these grid cells that contain hair geometry are returned and the sections of hair within the cell are independently checked to see if they fall within the area of influence, see Figure 3.

4.3 Multi-Resolution Simulation with the Grid

The grid aids us to localize our simulation on the areas of highest importance to the user. These areas are defined based on their distance from the viewer, visibility, motion, and the user's interaction with the hair. We adopted the method for choosing a level of detail based on distance and visibility created previously by [11], but have used our grid-based system to expand on the motion criteria and to include the user's interaction with the hair.

The motion of the hair is highly pertinent to the amount of detail needed to simulate the hair. Most applications performed on hair are localized to a small portion of the hair; the majority of the hair thus lies dormant. The sections of hair that are dormant are modeled with a lower LOD representation and resolution, determined by comparison against velocity thresholds, but we have gone a step further by effectively "turning-off" simulation for areas where there is no activity.

Each grid cell keeps track of the activity within the cell, tracking the hair sections that enter and exit the cell. When the action in a given cell has ceased and the hair sections in the cell have a zero velocity, there is no need to compute dynamic simulation due to gravity, spring forces, or collisions. The positions of the hair sections are thus frozen until they are re-activated. The cell is labeled as dormant and does not become active again until either the user interacts with the cell or until a new hair section enters the cell.

Rapid determination of the active cells and hair sections allows us to place the computational resources towards dynamic simulation for the hairs of highest interest to the user.

4.4 Real-Time Rendering

In order to guarantee a convincing and interactive experience for the user, our rendering algorithm has to run in real time and produce realistic images at the same time.

We implemented two separate specular highlights, due to the multiple modes of scattering inside and on the surface of the hair fibers observed by Marschner et al. [6]. We compute both specular terms by shifting the hair tangent in Kayija's original formulation [3] towards the hair root and towards the hair tip applying separate falloff exponents. The shifted tangents are used in the formulation proposed in [2]. All operations were performed in a fragment program for efficiency.

Realistic hair self-shadowing effects are hard to implement efficiently due to the large amount of dynamic geometry and the fineness of the hair strands. Our self-shadowing algorithm is based on opacity shadow maps created by [4] and recent GPU features [7]; it generates 16 opacity shadow maps in only one pass with multiple render targets, plus an extra pass for the actual rendering, without depth ordering required. As in [4], the opacity maps are placed at uniformly sampled distances from the eye, orthogonal to the view direction. Each of the four render targets holds four opacity maps, one in each 16-bit floating point component.

5 USER INTERACTION AND APPLICATIONS

Given our system for simulating and rendering hair described in the previous section, a user can now directly interact with hair through the 3D user interface and use operations commonly performed in hair salons. The operations supported in our system are described in this section.

5.1 Hair Cutting

Cutting hair is crucial for changing a hair's style, see Figure 4. Our cutting method models cuts performed with scissors. We model all the features of the cut, including capturing the hair that falls away or that is "cut-off". The location for cutting is defined by a triangle formed by the space between the open blades of scissors. Hair skeletons that intersect this triangle are then cut. At the cutting location, the skeleton S is split into two separate skeletons, S_1 and S_2; S_1 remains attached to the scalp, while S_2 falls down.

At the intersection of skeleton S and the cutting triangle, two new control points are created. One control point becomes the last point of skeleton S_1, while the second becomes the first point of S_2. The geometry of the fallen hairs remains consistent with the geometry of the hair below the sever point before the cut is performed; curliness, wetness, hair distribution and other properties are maintained in the fallen hair segments.

Figure 4: Example of haircutting, far right shows final style.

5.2 Applying Water, Hairspray and Mousse

Wet hair is modeled using the technique described in [9]. When water is applied to the hair, the mass points of the global-skeleton become heavier with the mass of the water. The overall motion of the hair is limited due to the extra weight and if the hair is curly, the global-skeleton will stretch under the extra weight and the curls will lengthen as expected. The volume of the hair in the areas where water is applied is decreased by constricting the radius of the current hair representation (strand grouping, cluster, or strip); these hair segments are then rendered to show the wet appearance as described by [9].

Hairspray is simulated on the hair by increasing the spring constants of the global-skeleton where it is applied. Moreover, *dynamic bonds* [9] are added between sections of hair that are in contact when the hairspray is applied. Dynamic bonds are extra spring forces that model the adhesive quality of hairspray to make the hair move as a group throughout subsequent motions.

We have chosen to model hair mousse that adds volume to hair. We inject volume into the hair model by growing the radii of the hair sections it affects. This process makes the hair fuller without adding more strands or skeletons.

5.3 Grabbing and Moving Hair

Typically when hair is clasped in the real world, a group of strands are selected at a local point along the strands. The user presses the stylus button and the control points that fall within the cone of influence are decided. Among these selected control points, only one control point per dual-skeleton is permitted to be *grabbed*. If multiple control points of a single dual-skeleton fall within the cone, the point that comes closest to the cone's center will be chosen.

In the grabbed state, as the user moves the stylus, the grabbed-point will follow the motion of the stylus. A grabbed-point cannot be pulled beyond its normal reach span (decided by its position in the dual-skeleton). The length of the hair is always maintained so that the lengths above the grabbed-point and below it are of consistent lengths while the point is moving. When the user wishes to release the grabbed-point(s), he or she releases the button of the stylus and the former grabbed-points will fall due to gravity.

5.4 Hairdryer

Hairdryers are one of the most common tools in a hair salon. When the stylus button is pressed, a strong constant force is applied in the direction of its orientation. Any control points that fall within the cone of influence receive this strong force. Moreover, if a wet control point (see Section 5.2) is influenced by the hairdryer, the control point will "dry"; the amount of water will decrease over the length of exposure dependent on the strength of the hairdryer force.

6 RESULTS AND DISCUSSION

Our virtual hairstyling system demonstrates the usefulness of our multi-resolution techniques for interactive hair modeling and was implemented in C++. The initial hairstyles are loaded as a preprocess.

Figure 5: Comparison between real (top) and virtual use of common hair salon activities (left) applying water (right) blow-drying hair.

We have been able to allow physically-based user interaction with dynamic hair while modeling several common hair salon applications. Figure 5 shows a comparison of real hair under the influence of common hairstyling applications with virtual hair styled by our system under the same conditions. Level-of-detail representations coupled with our simulation localization scheme have accelerated the animation of hair so that a user can actually interact with it.

Dynamic simulation, including implicit integration, LOD selection, hair applications (wetting, cutting, etc.), and collision detection, to create the hair model shown in Figure 1 ran at an average of 0.092 sec/frame. This figure comprised between 37 to 296 skeleton models, determined on-the-fly throughout the simulation, with an average of 20 control points each. At the finest resolution, the model contained 8,128 rendered strands; throughout the simulation the rendering LOD contained between 6K and 1,311K rendered vertices. Lighting and shadow computations on the GPU were performed in 0.058 sec/frame on average. The benchmarks were measured on a desktop PC equipped with an Intel® Xeon^{TM} 2.8Ghz processor with 2.0 GB RAM and an NVIDIA® GeForce^{TM} 6800 graphics card.

7 SUMMARY

We presented a prototype VR system involving an intuitive 3D user interface and methods for animating and rendering hair that allows for a user to interactively manipulate hair through several common hair salon applications. This system provides a level of user interaction that has been previously too complex to achieve. We are interested to explore the use of two-handed haptic gloves with our system to provide higher fidelity force feedback to the user, while allowing for further interaction capability and the creation of even more complex hairstyles. User studies involving professional stylists and novices on using this VR salon system would help to determine the effectiveness of this work.

REFERENCES

[1] F. Bertails, T-Y. Kim, M-P. Cani, and U. Neumann. Adaptive wisp tree - a multiresolution control structure for simulating dynamic clustering in hair motion. In *ACM SIGGRAPH Symposium on Computer Animation*, pages 207–213, July 2003.

[2] W. Heidrich and H.-P. Seidel. Efficient rendering of anisotropic surfaces using computer graphics hardware. *Proc. of Image and Multi-dimensional Digital Signal Processing Workshop (IMDSP)*, 1998.

[3] J. T. Kajiya and T. L. Kay. Rendering fur with three dimensional textures. In *SIGGRAPH '89: Proceedings of the 16th annual conference on Computer graphics and interactive techniques*, pages 271–280. ACM Press, 1989.

[4] Tae-Yong Kim and Ulrich Neumann. Opacity shadow maps. In *Proceedings of the 12th Eurographics Workshop on Rendering Techniques*, pages 177–182. Springer-Verlag, 2001.

[5] N. Magnenat-Thalmann and S. Hadap. State of the art in hair simulation. In *International Workshop on Human Modeling and Animation*, Korea Computer Graphics Society, pages 3–9, June 2000.

[6] Stephen R. Marschner, Henrik Wann Jensen, Mike Cammarano, Steve Worley, and Pat Hanrahan. Light scattering from human hair fibers. *ACM Trans. Graph.*, 22(3):780–791, 2003.

[7] NVIDIA. Technical report, NVIDIA, 2005. http://www.nvidia.com/.

[8] E. Plante, M-P. Cani, and P. Poulin. Capturing the complexity of hair motion. *Graphical Models (Academic press)*, 64(1):40–58, January 2002.

[9] K. Ward, N. Galoppo, and M. C. Lin. Modeling hair influenced by water and styling products. In *International Conference on Computer Animation and Social Agents (CASA)*, pages 207–214, May 2004.

[10] K. Ward and M. C. Lin. Adaptive grouping and subdivision for simulating hair dynamics. In *Pacific Graphics Conference on Computer Graphics and Applications*, pages 234–243, October 2003.

[11] K. Ward, M. C. Lin, J. Lee, S. Fisher, and D. Macri. Modeling hair using level-of-detail representations. In *International Conference on Computer Animation and Social Agents*, pages 41–47, May 2003.