A Hybrid Image-CAD Based System for Modeling Realistic Hairstyles

Xuan Yu,* Zhan Yu,† Xiaogang Chen,‡ Jingyi Yu§
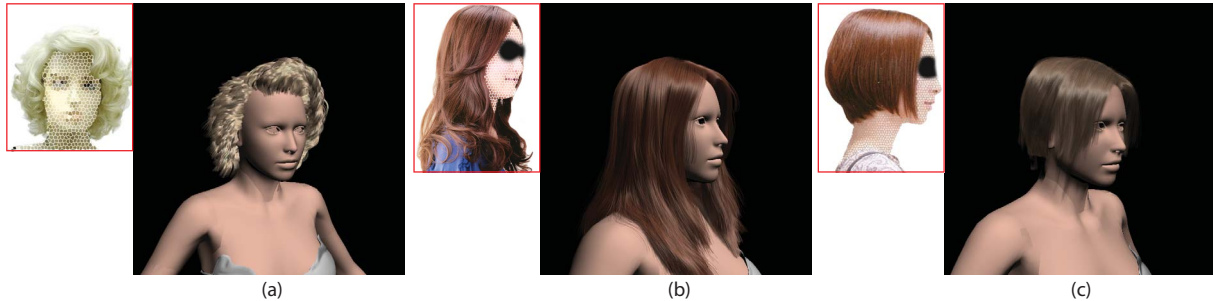
Figure 1: *A Marilyn Monroe style, long wavy hair, straight short hair, with their input photos.*

Abstract

Generating realistic hairstyles is critical in modeling human characters in computer games and feature films. State-of-the-art geometric modeling tools demand extensive user efforts and artistic skills. Image-based approaches attempt to automate the modeling process but require using special acquisition systems and/or computer vision techniques. In this paper, we present a new hybrid image-CAD based hairstyle modeling tool. We use the rich resource of hairstyle photos from fashion magazines as inputs where each hairstyle is commonly captured with a high quality and from multiple ($2 \sim 3$) viewpoints. We develop a sketch-based interface to allow users to draw a small number of strokes on different views and then map them to a sparse set of 3D splines. We then connect a web of splines as a guide hair surface and use it to compute a 3D orientational field within the hair volume for growing hair fibers. To further refine hairstyle models, we provide a set of tools for adjusting hair fiber length and density, perturbing hair fibers, creating wisps and curls, etc. Finally, we integrate an interactive rendering engine with the modeling modules so that the user can assess the model under realistic shading. Compared with existing solutions, our system significantly reduces user efforts and produces high quality hairstyle models consistent with the underlying hairstyle in the photo.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

Keywords: Image based hair modeling, hair mesh, volume modeling

---

*Qualcomm, e-mail:xuanyuchina@gmail.com
†University of Delaware, e-mail:yshmzhan@udel.edul
‡Shanghai Jiaotong University, e-mail:cxg@sjtu.edu.cn
§University of Delaware, e-mail:yu@eecis.udel.edu

1 Introduction

Hairstyles play a critical role in producing realistically looking human characters in computer games and feature films. They are markers and signifiers of culture, fashion, social class, age, gender, etc. Fig.1(a) shows the iconic pin curl hairstyle of Marilyn Monroe which is easily recognizable even on an avatar. Creating complex hairstyles as such however remains an open problem in computer graphics. Even with advanced modeling tools such as Autodesk Maya and 3D Studio Max, building realistic hairstyles from scratch is an "hairy" task: the users need to have both artistic skills and geometry backgrounds. For example, to produce the hair model and the animation of Rapunzel in the film "Tangled", Disney Animation Studio assembled a team of more than 30 animators and software engineers "that Vidal Sassoon himself would envy" [Keegan 2010]. Further, the modeling process is highly time consuming and multiple iterations for refinements are needed. For example, a single scene in Tangled can "take weeks for computer to generate" [Keegan 2010].

The challenges in generating realistic hairstyle models are multifold. First and foremost, hair is difficult to model. A realistic hair model contains a large number of fibers, generally in the magnitude of tens of thousands. Second, the geometric structure of hair fibers can vary significantly in race, geography and culture. Modifying hair fibers using CAD tools is too time consuming. Computer vision approaches can be potentially used to automatically generate hairstyles. However, most existing solutions require sophisticated setups (e.g., a large number of views, special lighting, etc). Finally, it is difficult to determine the quality of a hairstyle model without rendering it at near photo-realistic quality. Human eyes are sensitive to slight shading errors such as the missing of self-shadowing or incorrect specular highlights as we observe hair in daily life. A good hair modeling engine hence should provide interactive visual feedbacks, which has been largely missing.

In this paper, we present a hybrid image-CAD based solution for creating realistic hair models. Different from previous image-based solutions, we use hairstyle photos obtained from hair fashion magazines, an abundant and easily accessible resource from the internet. These images were captured by professional photographers and cover most popular hairstyles. Each hairstyle is usually captured from different viewpoints. Our system provides a four-view interface (front, side, back, and 3D) to assist the user to interactively design the guide surfaces (Fig. 2). Specifically, the user roughly draws strokes on 2D views and our tool automatically aligns the strokes with the orientation field of the image and creates a sparse
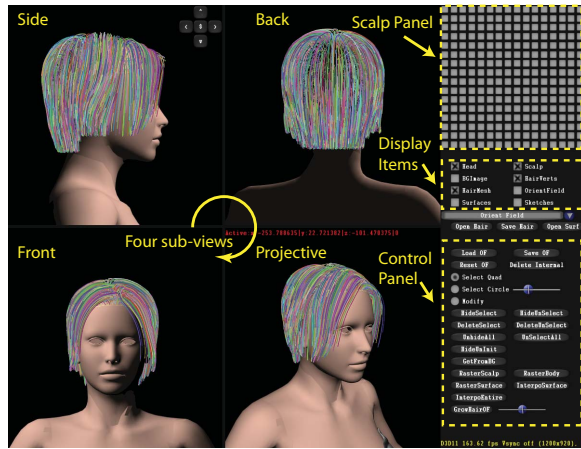
Figure 2: *The user interface of our system. We provide four views (front, back and side and a 3D perspective view) for the user to see and modify the model. We further provide a panel on the right side of the window with necessary tools to control and operate on hair.*

set of 3D curves. We connect these splines to generate the guide hair surface(s). To synthesize hair fibers, we map the guide surfaces, the scalp/body geometry, and the smoothness as constraints and solve for the 3D orientation field [Yu 2001] within the hair volume and grow hair fibers along the orientation fields.

Our system also provides an additional set of tools for refining hairstyle models. These include simple operators such as adjusting fiber length/density and more complex ones such as perturbing hair fibers and creating wisps and curls. Finally, we integrate an interactive rendering engine with the modeling modules. It allows the user to quickly assess the quality of the model for further refinements. Our rendering engine adopts the recent K-buffer hair rendering solution [Yu et al. 2012] to simultaneously produce volumetric shadows, transparency, and anti-aliasing. We demonstrate using our system for creating a range of hairstyles. In nearly all examples shown in the paper, we use no more than 20 splines as user input. Yet the resulting models appear realistic and consistent with hairstyle in the input photos.

2 Related Work

We generally categories hair style modeling tools into four categories, a comprehensive study of hair modeling can also be found at [Ward et al. 2007a].

CAD Based. Approaches in this category focus on designing effective geometric representations such as strip-based representations [Koh and Huang 2000; Koh and Huang 2001], thin shell volumes [Kim and Neumann 2000], NURBS surfaces [Noble and Tang 2004; Sokol 2003], etc. These solutions have long served as a workhorse in computer games. However, the over-simplified hair geometry leads to noticeable artifacts when rendered. More complex geometry primitives such as wisps or generalized cylinders [Choe and Ko 2005; Xu and Yang 2001; Yang et al. 2000; Chen et al. 1999; Watanabe and Suenaga 1992] can be created by clustering hair fibers under user-specified parameters such as length distribution, deviation radius and strand-shape fuzziness. Multi-resolution representation [Wang and Yang 2004; Ward et al. 2003; Bertails et al. 2003; Kim and Neumann 2002] creates an initial hair model with only a small number of hair clusters and then adding details via subdivision. A notable advance is the recent HairMesh tool [2009] that has shown great success in modeling complex hair

models. By using polygonal mesh structures as geometric primitives, their system transforms hair modeling to a regular surface modeling procedure with full user controls. However, similar to hair modeling procedures in standard CAD tools(Maya, 3DS Max), HairMesh does not directly incorporate image guidance into the modeling process and its outputs rely on heavily on the artistic and modeling skills of the users.

Sketch Based. Sketch based approaches provide convenient user interfaces for modeling hair geometry [Zeleznik et al. 1996; Igarashi et al. 1999]. Mao et al. [2002; 2004; 2005] used boundary curves on the scalp surface to define the region for growing hair where a partition line on the scalp surface guides hair strands partitioning near the scalp and a silhouette line of the target hairstyle guides hair growth. [Malik 2005] developed a sketching interface through which the user can freely implant hair clusters and style the hair using freeform strokes. Fu et al. [2007] built the interface on top of 3D vector field representations [Yu 2001] where the user draws a sparse set of 3D curves to define the global shape of the desired hairstyle. Wither et al. [2007] use sketches to infer the geometric and mechanical properties of hair strands such as length, stiffness and curliness. These approaches require substantial user inputs to produce reasonable hair shapes. Our approach investigates how to facilitate this process by using hairstyle photos as guidance.

Physical Based Approaches. A fully automatic hair model tool is to use physical-based solutions. Resenblum et al. [1991] simulated the structure and dynamics of human hair using a mass/spring/hinge system. Anjyo et al. [1992] used a cantilever beam simulation for calculating hair bending. Chang et al. [2002] used a sparse set of guide hair for creating a series of rigid multi-body open chains. Bando et al. [2003] further simplified the hair geometry as a set of particles representing the hair volume density. Ward et al. [2007b] developed a virtual hair salon system where the user can directly emulate spraying, grabbing and cutting operations via a 3D haptic interface. Physical based hair modeling techniques generally require few manual interactions but are comparatively expensive. Further, the hairstyles that can be modeled with physical simulation are still limited.

Image Based Approaches. An emerging class of automatic or near-automatic hair modeling method is image-based modeling. Nakajima et al. [1999] constructed the hair bounding volume from the images and synthesize hair fibers within the volume. The results preserve global shape but the detailed geometry can greatly deviate from the original images. Grabli et al. [2002] adopted photometric approaches by analyzing images captured under controlled lighting conditions to recover only a sparse subset of the original hair from the images. To capture dense hair strands, Paris et al. [2004] presented an effective image filtering technique to first produce reliable 2D orientations on each input image and then recover rough hair geometry. They later proposed to construct a novel acquisition effective system [Paris et al. 2008] using multiple cameras and projectors. This solution is able to capture highly complex hairstyle with concavities and curls. Wei et al. [2005] captured hair under uncontrolled illuminations with a hand-held camera and synthesized hair fibers from local image orientations. Jakob et al. [2009] have recently developed a method to measure detailed fibers within a hair assembly. They capture hair images with a shallow depth-of-field and then reconstruct hair fibers at each in-focus depth layer. Most of these approaches require using special imaging rigs to capture images at many ($30 \sim 40$) desired angles/depths. Recently, Chai et al. [2012; 2013] developed single-view hair modeling approach which automatically generates 2.5D hair that can be rerendered faithfully to original image in near-frontal views. Our goal is to model complete 3D hair that is visually pleasing but no necessarily faithful to input in details. Our input images mainly serve as guidance for easier and more efficient modeling.
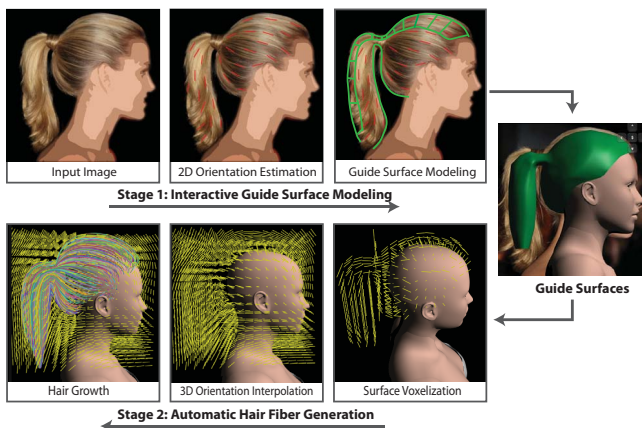
Figure 3: *Our hair modeling module consists of two stages: an interactive guide surface modeling stage and an automatic 3D hair fiber generation stage. The guide surface modeling stage uses user inputs to create guide surface set. The hair fiber generation stage then construct a 3D orientation field [2001] from the surfaces and grow hair fibers along the orientation field.*

Figure 4: *An elliptical bounding surface is used to approximate the geometry and the orientation field for median-straight hairstyles. We can combine multiple surfaces to generate more complex styles such a long wavy hairstyle. We use a vertical planar surface to create a mohawk hairstyle.*

We combine the advantages of image-based and CAD-based approaches by only using less then 3 images captured from different poses. These images are very common in hairstyle magazines. Fig.3 shows our modeling processing that contains two major stages: 1)interactive guide surface modeling and 2)automatic hair fiber generation. A guide surface is a two dimensional manifold within the 3D hair volume and is associated with an additional orientation vector field. It represents a major orientation feature of a hairstyle while geometrically simple enough to create and modify. For example, the shape of a median straight hairstyle shown in Fig.4(a) can be well described by an elliptical bounding surface whose orientations point downwards along the hair fibers. A more complex hairstyle such as a long wavy hair with side swept bangs in Fig.4(b) is relatively difficult to describe with a single surface. We thus use multiple surfaces, each representing a subset of the hair fibers (hair in front of the shoulders and within the forehead bangs, etc). A mohawk hairstyle Fig.4(c) can be effectively described by a simple vertical plane along the center. Our guide surface modeling is simple and flexible as they serve only as constrains of the global hair orientation field rather than be used the final output as most CAD tools did.

3 Interactive Guide Surface Modeling

The first stage in our modeling process is creating guide surfaces from user inputs. Similar to popular surface modeling techniques, we can benefit from multi-view reference images.

Reference Images. Hairstyling has been an important aspect of personal grooming, fashion and cosmetics. Popular search engines easily provide quality images of most hairstyles we may need. Better image sets even come with multiple views: front, side, and back. However internet resources or self portrait photographs do not come with camera poses, lighting consistency, and number of views. Therefore, fully automatic 3D reconstructions from these input images are difficult. Our system loads a set of (in most cases 3) photos of the target hairstyle as the reference views. The user can then directly operate (e.g., drawing splines) on them. We add the fourth 3D view to allow the user to tentatively view the model at arbitrary viewpoints. The user manually align them with a canonical 3D head/scalp model and the target hairstyle is modeled by using
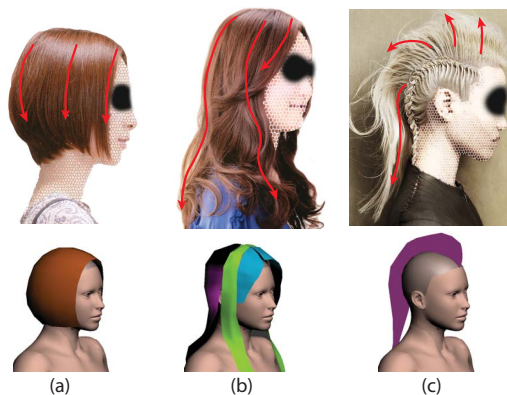
all views.

Surface Modeling Once images are properly aligned with the head model, we start creating 2D splines following the hair fibers in images. Like most image-based system, we provide automatic 2D orientation field extraction for each input image. Since the input hairstyle images contain many high frequency features (hair fibers) and are often scanned by low quality scanners, simple image processing operators such as Sobel lead to large errors. More sophisticated filtering schemes [Paris et al. 2004] can be used, but they tend to be slow. We adopt a simple and efficient scheme based on evaluating the gradient field over a $M \times N$ neighborhood of a given 2D location [Kass and Witkin 1987].We allow users to draw control points along a fiber and then reconstruct the fiber using splines that align with the orientation field in each image. We use the 2D orientation vector at each control point as the tangent directions and construct consecutive cubic splines. In case the user do not want to follow the extracted 2D orientation, e.g. local curly details are not required in this stage, he can choose Catmull-Rom spline which does not require local tangents. This gives the user full freedom to achieve desired spline shape.

By default, each control point on the 2D spline intersects with the head model surface in its z direction to get its 3D location. The
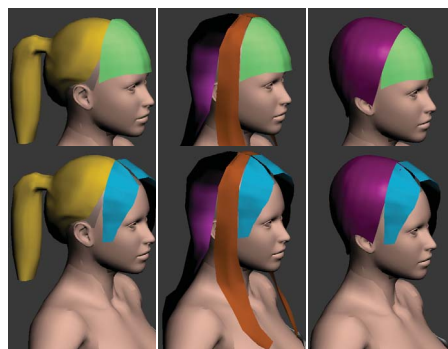

Figure 5: *Using our system, we can generate a class of hairstyle templates and then combine these templates to generate more variety of styles.*
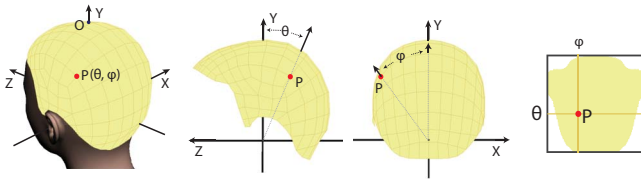
Figure 6: *2D parameterizations of the scalp surface under the spherical coordinate system.*

Figure 7: *We use the same guide surfaces for interpolating the hair fibers (a). By applying different textures to control the length of the fibers, we create a mohawk hairstyle (b) and a bald-looking hairstyle (c).*

user can make further adjustments to its 3D position in other views to match the reference image as well. Once a set of key splines are created, we connect them to form a guide surface. For a pair of neighboring splines we connect their corresponding control points by using Catmull-Rom splines. Notice that the resulting guide surface models not only provide shapes but also their orientations of the hair styles. The orientation field of the guide surface will be coherent with ones of the input images.

Template Guide Hair Surfaces. We further generate a library of template guide surfaces of popular hairstyles. Template surfaces, once built, can simplify the modeling process. For instance, the shape of the front part of hair (often referred to as fringe or bangs) is an important element in a hairstyle and different bangs can be shared between hairstyles. Using our representation, the user can create a set of template surfaces, each depicting a different bang style as shown in Fig. 5.

4 Automatic Hair Fiber Generation

Next, we generate 3D hair fibers from the guide surface.

Volume Discretization. We start with generating the 3D orientation field for a hairstyle. The user first specifies an axis-aligned bounding volume large enough to cover all hair fibers and discretize it at a specific resolution. Higher resolutions have the potential to create more detailed hair fibers but are also more expensive to compute. In the examples shown in the paper, we set the size of the cell/voxel to be $1cm \times 1cm \times 1cm$ as a balance between quality and speed. To further accelerate processing, we prune voxels that do not contain any hair fibers using a set of voxel pruning tools:

- **Select.** The user can select rectangular or spherical ranges of cells. The selected ones will be highlighted.
- **Hide/Unhide.** The user can hide the selected/unselected voxels. For example, when users need to work only with a specific set of voxels, they can first hide the rest of the cells to avoid undesired modifications. Once finished, they can use the "Unhide All" operation to show the hidden cells.
- **Delete.** The user can delete the selected or unselected cells. Combining the Hide/Unhide and Delete tools, the user can easily carve the orientation field to create a rough hairstyle.
- **Undo.** Undo the previous operation.
- **Reset.** Restore to the original bounding volume.

Voxels lying inside the 3D head and body models are also pruned.

Surface Voxelization Next, we map guide surfaces created by our system onto voxels. We first convert the spline surface to a triangle mesh and voxelize each triangle. This is analogous to the 2D triangle rasterization. Specifically, we first compute the 3D axis-aligned bounding box (AABB) of a triangle and then go through each cell inside the AABB and test if the cell overlaps with the triangle. To do this, we implement the classical 3D triangle-box inter-
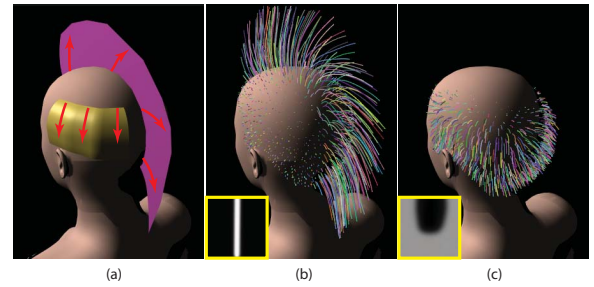
section test algorithm based on the separating axis theorem (SAT) [Eberly 2001; Gottschalk et al. 1996].

3D Orientation Field Estimation. Before we start integrating the gradient field to form hair fibers, we add a set of additional constraints:

- (1) **Scalp constraint:** All hair fibers should grow from the scalp surface along its normal direction. Therefore the orientation vector at each scalp voxel should be the scalp normal $\vec{n}$.
- (2) **Skin constraint:** Hair fibers touching the face or the body should be tangent to local surface. We assign the orientation at these voxels as $\vec{t} = \vec{y} \times \vec{n}$ where y is the gravity and $\vec{n}$ is the surface normal.
- (3) **Smoothness constraint:** Nearby hair fibers should have similar orientations.

We encode the first two constraints as bounding conditions and the last one using the 3D Laplacian operator. This results in a 3D Poisson system. We use the open source sparse matrix solver TAUCS library to solve for the 3D orientation field.

Hair Growth. Once we form the complete 3D orientation field, we grow hair fibers from it. We first sample hair fiber follicles on the scalp surface. We parameterize the scalp surface under the spherical coordinate system (Fig.6). Notice that one can distribute samples on the scalp surface in various ways, e.g., uniform vs. random vs. jittered sampling. The comparisons are shown in Fig.8. In our experiments, we find jittered sampling usually performs the best by redefining regularity while maintaining coherence. The 2D parametrization also allows us to assign texture maps to hair fibers to control fiber variables. As shown in Fig.7, by applying different textures to control the length of the hair fibers, we can generate significantly different hairstyles: a mohawk style Fig.7(b) and a bald style Fig.7(c).

We then iteratively grow each hair strand starting from its root follicle P_0 on the scalp surface. Assume that at the ith iteration, the hair strand has grown to position P_i that lies within voxel V_i. Notice that P_i does not necessarily lie at the center of the cell and therefore we compute the weighted average of all orientations within the 3×3 neighboring of V_i and use it to advance the strand by distance t, where t is half the cell height in our implementation.

5 Editing and Styling

Our guide surface based approach is suitable for modeling the global shape of a hairstyle. To enable more sophisticated styling,
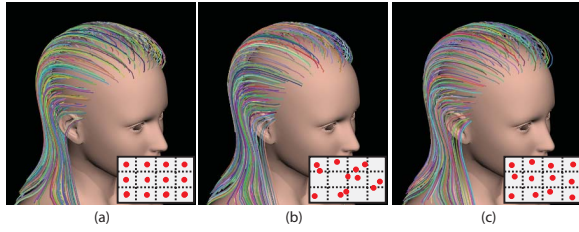
Figure 8: *We grow hair fibers from the follicles on the scalp surface. The follicles can be generated using uniform sampling (a), random sampling (b) or jittered sampling (c).*

we further develop a class of editing and styling tools for refining the initial model.

Simple Edits.   For simple edits, we provide the user with the **selection** tool to select a cluster of hair fibers in a specific view. Simple **cutting and extending** operator can then be applied to change the length of selected hair clusters. We also provide a **density adjustment** tool to controlling the number of hair fibers within a specific region on the scalp surface. Finally, we allow the user to change the number of control points on fiber splines via a **fiber adjustment** tool.

Wisps.   Wisps are an important class of features that greatly enhance realism. We create wisps by grouping nearby hair fibers: given n hair fibers $f_1, ..., f_n$, we group them into k sets $s_1, s_2, ..., s_k$ via k-means, each mapping to a wisp. Specifically, for each cluster m, we compute the central fiber by averaging the corresponding control points of all fiber splines within the cluster. In case that the number of control points differs across fibers, our fiber adjustment tool will automatically interpolate the splines to ensure identify number of control points.

Perturbation.   In real hair styling process, a hair stylist often deliberately perturbs layers of fibers to create geometric details. Operations as such include combing, spraying and drying with the goal to generate variations and volumetric appearances of hair. Simulating these operations via physical based approaches can be computational expensive. We instead produce these styling effects by applying a force field. Our strategy is to deform each hair fiber using the elastic beam model [Hearn 1997]. Specifically, we approximate the fiber as a set of rigid beams with elastic hinges that correspond to the control points. We then propagate the deforma-
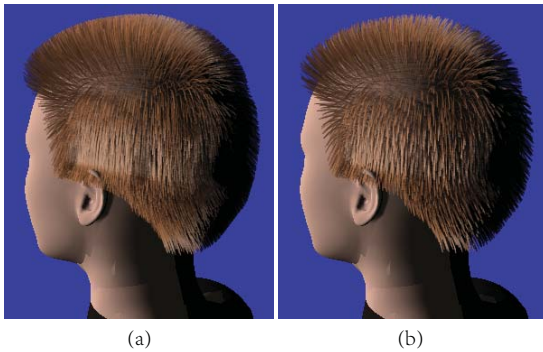

Figure 9: *The rendering results of hairstyles with (b) and without (a) wisp effects.*
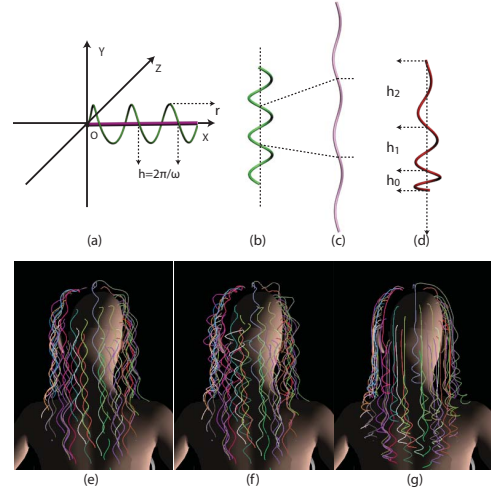

Figure 10: *Create curls and apply stretching. (a) shows the coordinate system used on each hair fiber for generating the offset function. (b) and (c) show the shape of the curls before and after stretching. The wave period of the curl($2\pi/\omega$) increases while the wave magnitude (curl radius) decreases. (d) The shape of the curls that are naturally stretched under gravity.(e) Curls produced by a constant wave function. (f) Curls produced by a wave function with random period and magnitudes. (g) Curls produced by our offset function that simulate the stretching of gravity.*

tion of a specific hair fiber from the root follicles. At each hinge, we compute the bending under the force field using the Momentum model [Daldegan et al. 1993] and rotate the rest of the fiber accordingly.

Curling.   To generate curls, we adopt an offset function on hair fibers for its simplicity, computational efficiency, and flexible controls. We parameterize the curl function along the tangent direction of the hair fiber (the X axis) as the helix function $P(x) = (x, r\sin(wx + \theta), r\cos(wx + \theta))$as shown in Fig.10(a). where w is the curl's frequency and r is the magnitude. This simple model can only produce uniform curls as shown in Fig.10(e). A simple fix is to apply random perturbations to w and t to generate more natural curls. However, the resulting curls appear irregular and unnatural as shown in Fig.10(f).

We observe that real curls are created by using uniformly sized rollers and then stretched (e.g., under gravity). When curls are pulled apart as shown in Fig.10(c), they spread wider but their sizes become narrower. We compute the stretch process as:

$$\int_0^{2\pi/w} \sqrt{((rsin(wx + \theta))')^2 + (rcos(wx + \theta)')^2 + 1}\, dx = 2\pi r_i$$
(1)

where the total length of a unit curl (integrated over the period of 0 to $2\pi/w$) remains as $2\pi r_i$, and $r = \sqrt{r_i^2 - \frac{1}{w^2}}$. Under stretch, a curl then becomes:

$$\int_x^l \sqrt{((rsin(wx + \theta))')^2 + (rcos(wx + \theta)')^2 + 1}\, dx = \frac{s}{w}$$
(2)

where the left side integration computes the length of the fiber from x to l and is proportional to the force that pulls the curl at x, s is the curl stiffness, and $\frac{s}{w}$ is the resisting force from the curl.

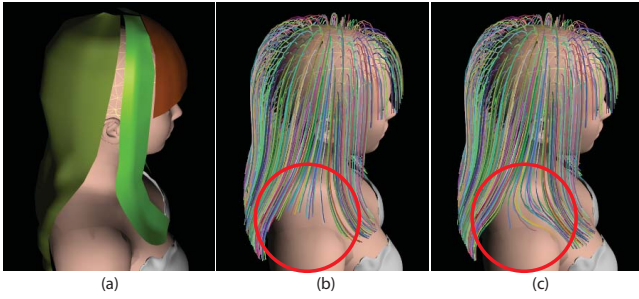Finally, we have $P(x) = (x, \sqrt{r_i^2 - \frac{(l-x)r_i}{s}}sin(\sqrt{\frac{s}{(l-x)r_i}}x +$

Figure 11: *Hair fibers generated from the guide surfaces (a) with (c) and without (b) using torso geometry as constraints.*

$\theta), \sqrt{r_i^2 - \frac{(l-x)r_i}{s}}cos(\sqrt{\frac{s}{(l-x)r_i}}x + \theta))$. We adjust curls using the radius of the roller r_i and the stiffness s as shown in Fig.10.

Real-time Feedback. An important component of our hairstyle modeling system is real-time visual feedback to allow users to assess the quality of the model. We adopt the recent GPU-based K-buffer solution [Yu et al. 2012] to simultaneously render volumetric self-shadow, transparency and anti-aliasing at interactive speeds. To briefly reiterate, we use a shadow-ray path to produce volumetric self-shadows and adds an additional view path to model light propagations from the hair volume to eye for producing transparency. To compute and accumulate the contribution of individual hair fibers along each ray (shadow or view) path, it then uses a K-Buffer [Callahan 2005; Bavoil et al. 2007] for storing the foremost K fragments along each ray for estimating shadow scattering and transparency scattering as well as for reducing aliasing.

We exploit the Append Buffer feature on the AMD radeon 5800 GPU with 2G video memory and use DirectX 11 Shader Model 5.0 as graphics API. The K-Buffer algorithms for computing volumetric shadow and transparency are implemented as the last stage of our rendering pass using a pixel shader. The rendering speed depends on the hairstyle model complexity. For most of our models that contain 10K to 50K fibers, we can render over 10 fps, thus providing interactive visual feedback for the user to assess the quality of the model.

6 Results and Discussions

As mentioned in Section 3, we have generated a small set of guide surface templates. Fig.4 (a) and (c) show two examples. The median straight hairstyles shown in Fig. 4(a) is modeled by an ellipsoid guide surface whereas mohawk hairstyles is modeled by a simple vertical plane guide surface. More complex hairstyles such as a long wavy hair with side swept bangs (Fig.4(b)) are difficult to model using a single guide surface but can be easily constructed using multiple guide surfaces. In Fig.5, we further combine a ponytail, a long and a median hairstyle with the same blunt bang and side-swept bang to generate three significantly different hairstyles. Most guide hair templates are of simple shapes and use a small number of sketch splines (about 20). Therefore, sketching and then generating these templates usually take less than 20 minutes.

Fig. 13 shows the hairstyles with hair fibers from the guide surfaces. From left to right, we show a ponytail, a median updo and a long straight hairstyle. For each style, we synthesize 30k hair fibers. Notice the spatial gap between two template surfaces in Fig. 13 (c) does not lead to missing hair fibers in between. This is because we use the guide surfaces as constraints and treat the gap as unknowns, and then interpolate the complete 3D orientation field (Section 4).
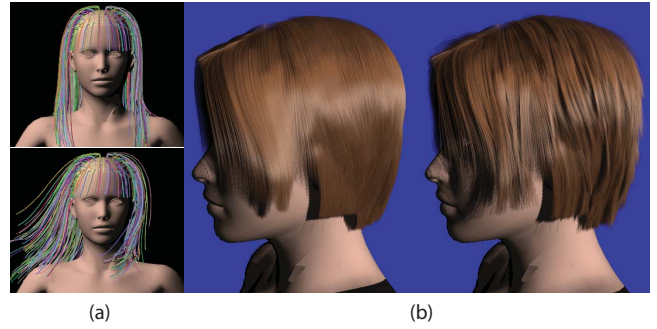

Figure 12: *Applying force field to hair fibers. (a) Generating hair styles under wind blows by using our force field model. (b) The rendered hairstyles before and after applying a random force field for adding local variations.*

The resulting hair strands also appear highly realistic as they follow the flow field defined by the guide surfaces. Fig 11 shows the importance of adding body geometry (head and torso) constraints. Without adding these constraints, the resulting hair fibers appear sticking into the shoulder (Fig. 11 (b)) whereas with the constraint the fibers follow the should geometry and appear more natural (Fig. 11(c)).

Fig.9 shows adding wisps a short hair style. Hair fibers without wisps appear overly uniform whereas the ones with wisps appear much more natural and realistic. Notice that the hair fiber geometry are similar in both cases and the differences will be difficult to perceive without applying shading. This illustrates the importance of having real-time, photorealistic rendering in modeling. Fig. 12 demonstrates the hairstyle before and after applying the force field emulated by constant (Fig. 12(a)) or random direction (Fig. 12(b)) forces. The hair fiber geometry before applying the force field is overly flat since the hair fibers are interpolated under the smoothness constraint. The perturbation tool greatly adds realism by revealing rich geometric details reflected under self-shadowing (Fig. 12(b)).

Fig.10 shows the effect of the curling operator on a straight hair model. We are able to effectively reproduce non-uniform looking curls caused by gravity. In contrast, curls produced by a constant wave function in Fig.10(a) appear Cartoon and fail to resemble the shape of natural curls. Results using a randomized w and r in

Figure 13: *Sample hairstyle modeling results (bottom row) using the guide surfaces (top row).*

Figure 14: *Rendering results of long curly hairstyle.*

Fig.10(b) on the other hand appear irregular and incoherent. We use low stiffness s so that the hair fibers are almost straight near the root due to the stretching of gravity. The final model (Fig.10(c)) and rendering results (Fig. 14) are close to curly hairstyles on photographs (Fig. 1). We also find that having the real-time rendering capability provides critical guidance to assess and modify the model. For example, Fig. 14 reveals the inner layer of hair fibers of the which will be difficult to perceive without rendering the transparency and shadows.

7 Limitations and Future Work

A major limitation of our guide surface model compared with other geometric representations such as HairMesh is that ours cannot handle complex topological structures such as knots (Fig. 15). It is our immediate future work to explore the possibility of combining the two approaches: we can first generate guide surface and then use the geometric operator of HairMesh to change its topology. Currently, the performance bottleneck of our system is the 3D orientation field estimation. We commonly discretize the volume at a fine level (e.g. with a resolution about $100 \times 100 \times 100$ voxels) and the computation can take up to a few minutes. A sparse matrix solver based on GPU can significantly accelerate our solution and be naturally integrated with the renderer.

Finally, we plan to extend our system to support non-photorealistic hairstyles. Previous systems [Noble and Tang 2004] used enclosing NURBS surfaces to model each hair clumps and then applied cartoon style shading to the NURBS surface for rendering hair. We can modify our system to achieve similar goals: we can grow only a sparse set of hair fibers and then use their hair clump geometry to create the final cartoon looking hairstyle. This can be added as a new styling operator. Finally, we will investigate implementing our framework as a plug-in to industry standard CAD tools.

8 Acknowledgements

References

ANJYO, K.-I., USAMI, Y., AND KURIHARA, T. 1992. A simple method for extracting the natural beauty of hair. *SIGGRAPH Comput. Graph. 26*, 2 (July), 111–120.

BANDO, Y., NISHITA, T., AND CHEN, B.-Y. 2003. Animating hair with loosely connected particles. *Comput. Graph. Forum 22*, 3, 411–418.

BAVOIL, L., CALLAHAN, S. P., LEFOHN, A., COMBA, J. A. L. D., AND SILVA, C. T. 2007. Multi-fragment effects on the gpu using the k-buffer. In *Proceedings of the 2007 symposium on Interactive 3D graphics and games*, I3D '07, 97–104.

BERTAILS, F., KIM, T.-Y., CANI, M.-P., AND NEUMANN, U. 2003. Adaptive wisp tree: a multiresolution control structure for simulating dynamic clustering in hair motion. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, SCA '03, 207–213.

CALLAHAN, S. 2005. *The K-Buffer and Its Applications to Volume Rendering*. Master's thesis, University of Utah.

CHAI, M., WANG, L., WENG, Y., YU, Y., GUO, B., AND ZHOU, K. 2012. Single-view hair modeling for portrait manipulation. *ACM Trans. Graph. 31*, 4 (July), 116:1–116:8.

CHAI, M., WANG, L., WENG, Y., JIN, X., AND ZHOU, K. 2013. Dynamic hair manipulation in images and videos. *ACM Trans. Graph. 32*, 4 (July), 75:1–75:8.

CHANG, J. T., JIN, J., AND YU, Y. 2002. A practical model for hair mutual interactions. In *Proceedings of the 2002 ACM SIGGRAPH/Eurographics symposium on Computer animation*, ACM, New York, NY, USA, SCA '02, 73–80.

CHEN, L.-H., SAEYOR, S., DOHI, H., AND ISHIZUKA, M. 1999. A system of 3d hair style synthesis based on the wisp model. *The Visual Computer 15*, 4, 159–170.

CHOE, B., AND KO, H.-S. 2005. A statistical wisp model and pseudophysical approaches for interactive hairstyle generation. *IEEE Transactions on Visualization and Computer Graphics 11*, 2 (Mar.), 160–170.

DALDEGAN, A., THALMANN, N. M., KURIHARA, T., AND THALMANN, D. 1993. An integrated system for modeling, animating and rendering hair. *Computer Graphics Forum 12*, 3, 211–221.

EBERLY, D., 2001. 3d game engine design-a practical approach to real-time computer graphics. 2001.

FU, H., WEI, Y., TAI, C., AND QUAN, L. 2007. Sketching hairstyles. In *Proceedings of the 4th Eurographics workshop on Sketch-based interfaces and modeling*, ACM, 31–36.

GOTTSCHALK, S., LIN, M., AND MANOCHA, D. 1996. Obbtree: A hierarchical structure for rapid interference detection. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, 171–180.

GRABLI, S., SILLION, F., MARSCHNER, S. R., AND LENGYEL, J. E. 2002. Image-based hair capture by inverse lighting. In *Proc. Graphics Interface*, 51–58.

HEARN, E. 1997. *Mechanics of materials: An introduction to the mechanics of elastic and plastic deformation of solids and structural materials*, vol. 2. A Butterworth-Heinemann Title.

IGARASHI, T., MATSUOKA, S., AND TANAKA, H. 1999. Teddy: a sketching interface for 3d freeform design. In *Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, ACM Press/Addison-Wesley Publishing Co., New York, NY, USA, SIGGRAPH '99, 409–416.

JAKOB, W., MOON, J. T., AND MARSCHNER, S. 2009. Capturing hair assemblies fiber by fiber. In *ACM SIGGRAPH Asia 2009 papers*, ACM, New York, NY, USA, SIGGRAPH Asia '09, 164:1–164:9.

KASS, M., AND WITKIN, A. 1987. Analyzing oriented patterns. *Computer vision, graphics, and image processing 37*, 3, 362–385.

KEEGAN, R., 2010. The hairy task of creating rapunzel in 'tangled', November.

KIM, T.-Y., AND NEUMANN, U. 2000. A thin shell volume for modeling human hair. In *Proceedings of the Computer Animation*, IEEE Computer Society, Washington, DC, USA, CA '00, 104–.

KIM, T.-Y., AND NEUMANN, U. 2002. Interactive multiresolution hair modeling and editing. *ACM Trans. Graph. 21*, 620–629.

KOH, C. K., AND HUANG, Z. 2000. Real-time animation of human hair modeled in strips. In *Computer animation and simulation 2000: proceedings of the Eurographics Workshop in Interlaken, Switzerland, August 21-22, 2000*, Springer Verlag Wien, 101.

KOH, C. K., AND HUANG, Z. 2001. A simple physics model to animate human hair modeled in 2d strips in real time. 127.

MALIK, S. 2005. A sketching interface for modeling and editing hairstyles. In *Eurographics Workshop on Sketch-Based Interfaces and Modeling*, The Eurographics Association, 185–194.

MAO, X., KASHIO, K., KATO, H., AND IMAMIYA, A. 2002. Interactive hairstyle modeling using a sketching interface. In *Proceedings of the International Conference on Computational Science-Part II*, Springer-Verlag, London, UK, UK, ICCS '02, 131–140.

MAO, X., KATO, H., IMAMIYA, A., AND ANJYO, K. 2004. Sketch interface based expressive hairstyle modelling and rendering. In *Proceedings of the Computer Graphics International*, IEEE Computer Society, Washington, DC, USA, CGI '04, 608–611.

MAO, X., ISOBE, S., ANJYO, K., AND IMAMIYA, A. 2005. Sketchy hairstyles. In *Proceedings of the Computer Graphics International 2005*, IEEE Computer Society, Washington, DC, USA, CGI '05, 142–147.

NAKAJIMA, M., MING, K., AND TAKASHI, H. 1999. Generation of 3d hair model from multiple pictures. *IEEE Computer Graphics & Applications 12*.

NOBLE, P., AND TANG, W. 2004. Modelling and animating cartoon hair with nurbs surfaces. In *Proceedings of the Computer Graphics International*, IEEE Computer Society, Washington, DC, USA, CGI '04, 60–67.

PARIS, S., BRICEÑO, H. M., AND SILLION, F. X. 2004. Capture of hair geometry from multiple images. In *ACM SIGGRAPH 2004 Papers*, SIGGRAPH '04, 712–719.

PARIS, S., CHANG, W., KOZHUSHNYAN, O. I., JAROSZ, W., MATUSIK, W., ZWICKER, M., AND DURAND, F. 2008. Hair photobooth: geometric and photometric acquisition of real hairstyles. *ACM Trans. Graph. 27* (August), 30:1–30:9.

ROSENBLUM, R., CARLSON, W., AND TRIPP III, E. 1991. Simulating the structure and dynamics of human hair: modelling, rendering and animation. *The Journal of Visualization and Computer Animation 2*, 4, 141–148.

SOKOL, A., 2003. Modeling hair and fur with nurbs.

WANG, T., AND YANG, X. D. 2004. Geometric modeling. Kluwer Academic Publishers, Norwell, MA, USA, ch. Hair design based on the hierarchical cluster hair model, 330–359.

WARD, K., LIN, M. C., LEE, J., FISHER, S., AND MACRI, D. 2003. Modeling hair using level-of-detail representations. In *Proceedings of the 16th International Conference on Computer Animation and Social Agents (CASA 2003)*, IEEE Computer Society, Washington, DC, USA, CASA '03, 41–.

WARD, K., BERTAILS, F., KIM, T.-Y., MARSCHNER, S. R., CANI, M.-P., AND LIN, M. C. 2007. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics 13* (March), 213–234.

WARD, K., GALOPPO, N., AND LIN, M. 2007. Interactive virtual hair salon. *Presence: Teleoper. Virtual Environ. 16*, 237–251.

WATANABE, Y., AND SUENAGA, Y. 1992. A trigonal prism-based method for hair image generation. *IEEE Comput. Graph. Appl. 12*, 1 (Jan.), 47–53.

WEI, Y., OFEK, E., QUAN, L., AND SHUM, H.-Y. 2005. Modeling hair from multiple views. *ACM Trans. Graph. 24* (July), 816–820.

WITHER, J., BERTAILS, F., AND CANI, M. 2007. Realistic hair from a sketch. In *Shape Modeling and Applications, 2007. SMI'07. IEEE International Conference on*, IEEE, 33–42.

XU, Z., AND YANG, X. D. 2001. V-hairstudio: An interactive tool for hair design. *IEEE Computer Graphics and Applications 21*, 36–43.

YANG, X. D., XU, Z., WANG, T., AND YANG, J. 2000. The cluster hair model. *Graph. Models 62*, 2 (Mar.), 85–103.

YU, X., YANG, J. C., HENSLEY, J., HARADA, T., AND YU, J. 2012. A framework for rendering complex scattering effects on hair. In *Proceedings of the ACM SIGGRAPH Symposium on Interactive 3D Graphics and Games*, ACM, New York, NY, USA, I3D '12, 111–118.

YU, Y. 2001. Modeling realistic virtual hairstyles. In *Computer Graphics and Applications, 2001. Proceedings. Ninth Pacific Conference on*, 295–304.

YUKSEL, C., SCHAEFER, S., AND KEYSER, J. 2009. Hair meshes. *ACM Trans. Graph. 28* (December), 166:1–166:7.

ZELEZNIK, R., HERNDON, K., AND HUGHES, J. 1996. Sketch: An interface for sketching 3d scenes. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, ACM, 163–170.

(a)   (b)   (c)

Figure 15: *Styles that our system cannot produce. (a) A hair braid with cross intersecting hair fibers, (b) a tightly folded hair knot, and (c) extremely curly hairstyle.*