To appear in the ACM SIGGRAPH conference proceedings

Capturing Hair Assemblies Fiber by Fiber

Wenzel Jakob Jonathan T. Moon Steve Marschner

Cornell University

Abstract

Hair models for computer graphics consist of many curves representing individual hair fibers. In current practice these curves are generated by ad hoc random processes, and in close-up views their arrangement appears plainly different from real hair. To begin improving this situation, this paper presents a new method for measuring the detailed arrangement of fibers in a hair assembly. Many macrophotographs with shallow depth of field are taken of a sample of hair, sweeping the plane of focus through the hair's volume. The shallow depth of field helps isolate the fibers and reduces occlusion. Several sweeps are performed with the hair at different orientations, resulting in multiple observations of most of the clearly visible fibers. The images are filtered to detect the fibers, and the resulting feature data from all images is used jointly in a hair growing process to construct smooth curves along the observed fibers. Finally, additional hairs are generated to fill in the unseen volume inside the hair. The method is demonstrated on both straight and wavy hair, with results suitable for realistic close-up renderings. These models provide the first views we know of into the 3D arrangement of hair fibers in real hair assemblies.

CR Categories: I.3.7 [Computer Graphics]: Three-Dimensional Graphics and Realism

Keywords: Hair, 3D scanning

1 Introduction

Rendering human hair involves many challenges. The 3D arrangement of fibers must be specified initially, animated through time, and rendered with simulations of local and global light scattering. Great progress has been made on all these fronts, and rendered hair is now routinely substituted for real footage in visual effects.

But rendered hair still cannot stand up in closeups, and one important reason is that the appearance of rendered hair depends strongly on how the fibers are arranged locally. Hair modeling systems create small-scale detail by applying random perturbations to fiber shapes, producing arbitrary arrangements of curves that do not much resemble the structure of real hair. Even with state-of-the art rendering methods, renderings of these models appear obviously different from real hair,

Research on modeling hair has primarily focused on the large scale structure, while research on rendering hair has assumed that fiber-accurate models will be produced by someone else. Part of the difficulty in improving the small-scale structure of hair models is that it's not known how fibers are arranged in real hair. Photographs clearly show that the fiber arrangements in rendered hair are wrong, but it is difficult to tell how to improve them by looking at images, in which the geometry mixes with many other confounding factors.

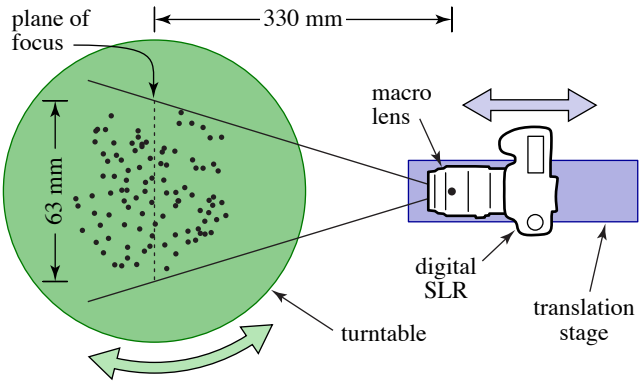

Figure 1: *Our measurement setup consists of a digital still camera imaging a sample of hair through a macro lens. The hair rotates around a vertical axis, and the camera translates along its optical axis, which is perpendicular to the rotation axis.*

This paper presents a method for capturing the small-scale structure of hair using large sets of macrophotographs that isolate individual fibers using shallow depth of field. We demonstrate a system that individually measures the shapes of hair fibers within a working volume about 6 cm across, providing new views into the small-scale structure of hair. Because of the size of the volume, and because the hair must remain still during a lengthy capture session, it is most readily applicable to disembodied hair switches. Many hair capture systems capture full hairstyles on live subjects, but only measure large-scale structure. Our system is complementary to these, aiming instead to recover accurate geometry on much smaller scales.

Apart from providing models that can be used in ground-truth verifications of hair rendering algorithms, the resulting datasets also offer new possibilities for improving the level of realism achieved by current hair modeling tools. An immediate application is the study of statistical properties found in the fiber distributions of different hair types, such that better models for randomly generating hair through existing large-scale structures can be found. Another application is to use the accurate fine-scale geometry as a source of detail transferable onto coarse hair models, which provides an interesting application of example-based hair modeling.

1.1 Method overview

Our measurement setup, diagrammed in Figure 1, consists of a digital still camera imaging a sample of hair through a macro lens. The hair is mounted on a turntable, with the rotation axis perpendicular to the camera's optical axis. The camera is mounted on a translation stage, with the translation axis parallel to the optical axis. For each of several orientations of the turntable, we capture images at a densely spaced set of camera positions, sweeping the focus plane through the hair volume.

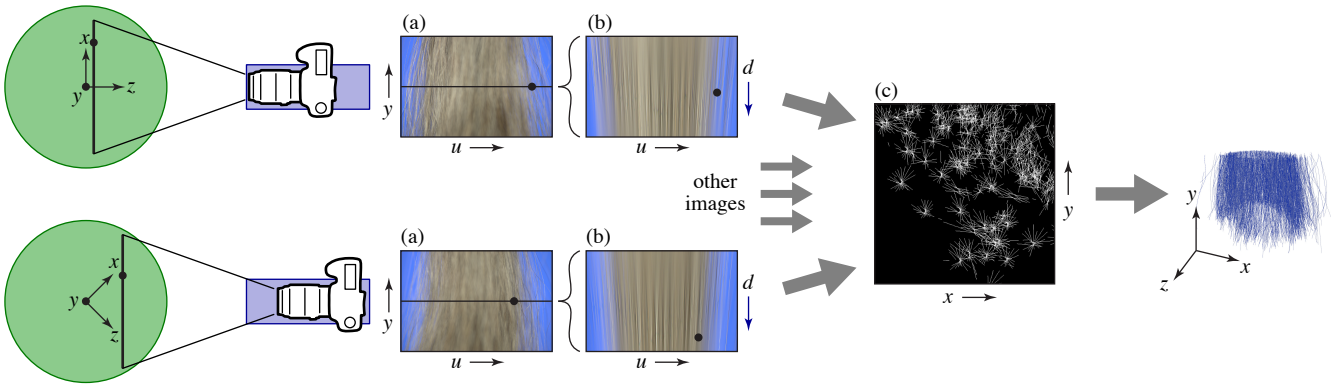

Figure 2: *Our capture proceeds by taking several sets of photographs, each consisting of a sweep of the camera along the translation stage. (a) Example images from two sweeps, seeing the same hair rotated to a different angle. (b) Images plotting u against camera position reveal the 3D locations of hairs. (c) Features located from many sweeps are combined, forming stars that reveal the positions of the fibers.*

The resulting images generally have only a few hair fibers in focus, and we know those fibers are very close to the plane of focus. Simply by observing a well focused image of a fiber, we know its position in 3D space. Two problems with this basic approach are the difficulty of determining exactly when a fiber is in focus, which leads to uncertainty in the fibers' positions in the direction of the translation axis, and the possibility for occlusion by other fibers. Both of these problems are alleviated by including translational sweeps captured with the turntable at a number of different rotations—in effect, from a number of camera positions spaced around the hair. Each sweep observes fibers from a different direction, and the observations of the same point have complementary uncertainties, together locating the fiber precisely enough to separate it from the other fibers in the assembly.

In the following sections are a survey of prior work and a more detailed explanation of the measurement technique, followed by the details of the data processing pipeline. Finally, we present results from three hair samples.

2 Prior work

Research on hairstyle generation has primarily focused on generating appropriate large scale structure under user control. This large scale structure is encoded as surfaces [Kim and Neumann 2000], guide strands, or wisps [Choe and Ko 2005], often arranged in multiresolution hierarchies [Kim and Neumann 2002; Bertails et al. 2003]. The structure is created through direct editing, surface scans [Sobottka et al. 2006], simulation of guide strands or wisps as beams [Anjyo et al. 1992; Choe and Ko 2005] or rods [Bertails et al. 2006], advection through vector fields [Hadap and Magnenat-Thalmann 2000; Yu 2001], or other means to allow smooth high-level control over the hair fibers. Ultimately, fibers are generated randomly to follow the large-scale structure, using sophisticated statistical distributions [Yu 2001; Choe and Ko 2005] that are tuned to achieve the desired appearance. Finally, different hairstyles can be combined through detail transfer operations [Wang et al. 2009]. For a more complete discussion of methods in hairstyle modeling, please refer to the survey by Ward et al. [2007].

Common to nearly all hair modeling systems is the random generation of the individual fibers—which are ultimately what is rendered—by random processes that have little to do with the properties of real hair. This is true even when the larger-scale structure is controlled by a physical simulation. We are just beginning to see simulations that treat the strands individually [Selle et al. 2008], and achieving realism with such simulations will require physical models that better reflect the properties of real hair fibers.

To render the fibers once they are generated, much recent work has focused on developing physically accurate transport simulations for hair [Moon and Marschner 2006; Moon et al. 2008; Zinke and Weber 2007; Zinke et al. 2008]. Moving from ad hoc diffuse components to methods accounting for multiple scattering makes a major improvement in the appearance of light colored hair, but the realism of the resulting images, particularly in close-ups, is severely limited by the quality of the fiber geometry. The results of this paper also aim to provide the first models for which renderings can be compared directly with photographs, to separate the limitations of the rendering method from the limitations of the model.

There are several existing methods that model hairstyles from photographs. Grabli et al. [2002] proposed the first technique to recover sparse hair fiber directions from images by using controlled light sweeps and knowledge of hair fiber scattering profiles. Paris et al. [2004] improved upon this technique by using more robust filtering methods and a means for combining results from several viewpoints. An even more flexible approach was later demonstrated by Wei et al. [2005] that allowed uncalibrated lighting and camera positions, and used triangulation rather than inverse lighting to estimate hair directions. And most recently Paris et al. [2008] demonstrated a system that uses a dome configured with multiple cameras, lights, and projectors to capture complex hairstyles.

Unfortunately, none of these existing methods is capable of recreating the actual individual strands of the input hair sample. Instead, their goal is to determine the volume of space filled with hair and then to grow curves into that space using chaining, hair interpolation, or advection methods. And while image-based rendering techniques can produce good appearance [Paris et al. 2008], it comes with important limitations compared to having realistic geometry. Our goal and contribution in this work is the ability to precisely capture the individual hairs.

Our processing uses two-dimensional Gabor filters, which are similar to the spatial and spectral sensitivity profiles of cells in the mammalian visual cortex [Jones and Palmer 1987]. Their optimal localization in both space and frequency have also made them popular in areas such as texture segmentation [Jain and Farrokhnia 1991] and fingerprint recognition. Here, Gabor filters are used to detect local ridge orientations, to which they have successfully been applied in the past, for example in fingerprint identification [Yang et al. 2003].

The idea of isolating layers, and "seeing around" occluding geometry, using a large aperture, is the standard practice in mi-

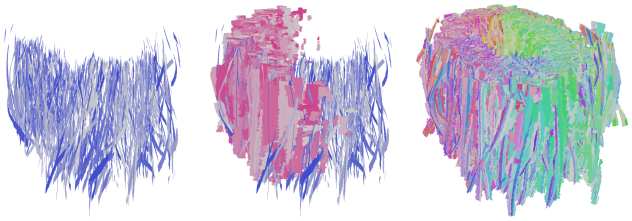

Figure 3: *After detected features have been rotated into a shared coordinate system, they form ribbons, whose intersections identify the path of individual hair fibers (1, 2 and 24 angles).*

croscopy [Inoué and Spring 1997; Levoy et al. 2006]. While our system comes nowhere near microscope-like apertures, being limited by the relatively large field of view, one could imagine using a higher magnification and a larger camera format to create an even shallower depth of field.

Because it uses multiple views, our method bears some resemblance both to stereo vision and triangulation scanning, though the particular methods we use are quite different because we are looking for isolated fibers rather than a smooth surface. In comparison to confocal stereo [Hasinoff and Kutulakos 2009], we retain all camera parameters throughout a measurement, which simplifies the task of transforming the captured data into a common coordinate frame. We borrow the idea of space carving from the scanning literature [Curless and Levoy 1996]. Another closely related ranging method is depth from focus [Ens and Lawrence 1993]; in essence, we use depth from focus to estimate fiber positions to disambiguate an otherwise hopeless correspondence problem between widely separated views of very messy geometry.

3 Measurement

Our measurement setup has the sample of hair, in our experiments a small switch of a few thousand fibers, suspended freely hanging from a post attached to a motorized turntable. The volume of the hair is roughly centered on the rotation axis. The camera (Canon EOS 5D) views the hair from a distance of 330 mm, and rides on a translation stage that moves the camera parallel to its optical axis. Illumination comes from a large area source behind the camera, made from an array of 14 linear fluorescent lamps, that subtends about 60 degrees horizontally and vertically from the hair. A blue background is placed behind the hair. There are several motivations behind the decision to use a large aperture. One advantage is that few features appear in any given image, making them easier to detect reliably. Furthermore, the shallow depth of field gives a depth estimate that is invaluable in establishing correspondence in messy geometry. And finally, the large aperture mitigates the effects of occlusion: because each pixel averages over a roughly 10-degree cone of rays passing through the focal point, a few hairs passing well in front of the hair under observation will not occlude it from view.

The key alignments in the system are that the translation and rotation axes are perpendicular, that the projection of the rotation axis is exactly vertical in the camera's view, and that the camera's plane of focus contains the rotation axis when the translation stage is at zero. These alignments are readily established by aligning a very flat reference grid with the rotation axis, then photographing it for a number of different rotations and translations.

The hair is viewed through a macro lens (Canon 100mm $f/2.8$[1]) at full aperture, producing an in-focus image of a rectangular area about 42 by 63 mm in size with an observed depth of field of 1.2mm. The camera has a color sensor 4386 pixels across, but we use only the red pixels from its Bayer array for feature detection, to obtain maximum hair brightness and separation from the blue background, resulting in half-resolution images 2193 pixels across. This provides 35 pixels/mm at the focus plane, sufficient to clearly resolve hairs (which are on the order of 100 μm in diameter).

Our measurement consists of sets of photographs, each taken with the hair at a fixed orientation as the camera makes a sweep along the translation stage. The camera positions are densely spaced (5/mm) over a 75 mm range, which is large enough to sweep the focus plane through the entire hair assembly. The sweep is repeated for 24 orientations of the turntable, so a full dataset comprises 9024 images (376 camera positions for each of 24 angles). With this setup, we are able to capture a vertical cylinder of radius 32mm and height 60mm. Larger volumes require sweeps along additional axes or, alternatively, a higher-resolution sensor. We label the sweeps by the orientation θ of the turntable, and label each image with the distance d from the center of rotation to the camera's plane of focus. For convenience, in this paper we express image coordinates (u, v) in terms of world distance measured on the focus plane. The projection of the rotation axis on the image plane is exactly vertical in the image, and we define its horizontal coordinate as $u = 0$.

Interpreting the data. We parameterize the hair volume (object space) with a cartesian coordinate system (x, y, z), where the y axis is the rotation axis. This coordinate system moves with the hair as the turntable rotates. For objects in focus, the y coordinate in object space is always the same as the v coordinate in image space, because the focus plane is parallel to the rotation axis and the motion of the camera is perpendicular to that axis.

In each sweep, the plane of focus will pass any point (x, y, z) on a hair fiber exactly once, at which time the fiber will in principle appear exactly focused, if it is visible to the camera (Figure 2a). If we denote the projection of that point in the camera image as (u, v), and take d to be the camera distance of the image where the point was most sharply focused, then x and z are related to u and d by a planar rotation:

$$\begin{bmatrix} u \\ d \end{bmatrix} = R(\theta) \begin{bmatrix} x \\ z \end{bmatrix},$$

where $R(\theta)$ is a counterclockwise rotation through the angle θ. The geometry underlying these relationships is shown in Figure 2.

In practice, the fiber appears in focus over a short range of camera positions, so although it is very precisely located (uncertainty less than 1/10 mm) in the u direction, its position is much more uncertain (uncertainty about 1 mm) in the d direction. Displaying the data for one row of the camera as an image, with horizontal coordinate u and vertical coordinate d (Figure 2b), displays the observation of a fiber as a short streak, reflecting this depth uncertainty.

By integrating observations from several sweeps (Figure 2c) we have the opportunity to observe each fiber from more than one direction, thereby tying its position down completely. Coordinating multiple observations is simple: the (u, v, d) data from each sweep simply needs to be rotated about the y axis to bring it into the (x, y, z) coordinate system. In an (x, z) slice, the set of observations of a point on a fiber will create a star of several intersecting streaks, and it is by locating these intersections of features that we extract 3D hairs from the volume.

When fitting 3D curves to our data, it is useful to think in terms of the whole dataset, rather than considering a slice at a time as Figure 2 does. In the object coordinates, each sweep covers a rectangular-prism-shaped volume defined by the two axes of the camera image and the direction of translation. Within this volume,

[1] We investigated but found no benefit in correcting for geometric distortion or field curvature.

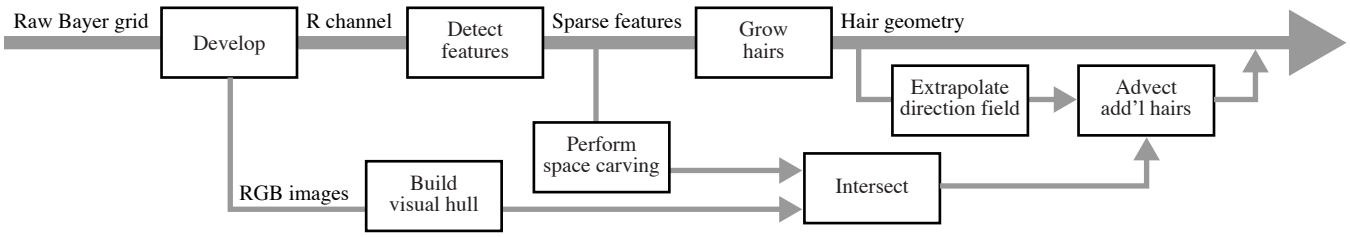

Figure 4: ***Overview****: A ridge detection stage identifies well-focused hairs in each of the input photographs, after which three-dimensional fibers are grown through the resulting feature data. To fill in empty areas, occluded regions are identified using a space carving approach and additional fibers are then advected through an extrapolated velocity field.*

the set of points where a hair appears focused is a "ribbon" that is always edge-on to the camera's view direction. Finding hairs in the volume amounts to finding sets of intersecting ribbons that are mutually consistent. Figure 6 illustrates this idea, which is central to the processing algorithm in the following section.

4 Data Processing

To sensitively and reliably fit curves to this data, the two keys to our approach are (1) to detect features in the images first, which vastly cuts down the volume of data so that the processing can be done in memory, and (2) to fit curves directly to the feature data, rather than building isolated 3D features and chaining them.

Figure 4 shows a high-level overview of data processing. After a full dataset has been acquired, our processing pipeline takes the high-resolution photographs and turns them into hair geometry in the form of finely tesselated line segments. This transformation proceeds in three stages: at first, each image is passed through a ridge detection operator, which detects both the location and image-space 2D orientation of well-focused hairs. The resulting sparse ridge data is passed to a fiber-growing algorithm that traces hair fibers in 3D space using observed position and orientation information from multiple angles. This step resolves many visible hairs, but leaves holes in regions where occlusions make detection impossible. To produce renderable geometry, we fill in these parts by growing additional hairs through an extrapolated fiber direction field. To ensure that these are only added in areas that are empty due to occlusion, we construct both a visual hull and a space carving volume to guide their placement. For simplicity, in this section, we will interpret the hair as being fixed, with the camera rotating to observe them from different directions.

4.1 Filtering

A typical measurement photograph consists mostly of hairs that are blurred over a large region due to their distance to the focal plane, while distinct bright lines indicate the positions of well-focused hairs. The main task of the filtering stage, then, is to pinpoint the location of these image-space ridges, while discarding hairs that are not completely in focus. To discern different hairs, we require a filter capable of turning ridges into continuous single-pixel wide features. Additionally, to establish correspondence between different photographs later on, good localization is another requirement.

Briefly ignoring that the Canny [1986] operator is in effect an edge detector, its underlying framework possesses both of these qualities. We use a modified version of the filter, which finds ridges in addition to image-space orientation information, while also preserving the localization and continuity properties.

As is done in the scale-space framework [Lindeberg 1998], we first convolve the input images with a Gaussian to detect features at a specific scale. This allows us to tune the filter to respond to features at roughly the size of a focused hair. Canonical implementations of the Canny operator then estimate horizontal and vertical image gradients using a pair of Sobel filter convolutions, after which the approximate edge orientation of a pixel is given by the arctangent of their ratio. While generally sufficient for detecting edges or ridges, the angular accuracy of this approach is too low for our purposes. Here, orientation information additionally serves as an input to a later stage of the algorithm, where higher accuracy will help to improve the quality of the reconstruction. It is therefore desirable to implement a more sophisticated orientation estimator.

A two-dimensional Gabor filter is constructed by modulating an oriented sinusoidal plane wave by a by a Gaussian envelope. The real component of an even-symmetric Gabor filter is given by [Jain and Farrokhnia 1991]:

$$G(u,v) := \exp\left(-\frac{1}{2}\left[\frac{\tilde{u}^2}{\sigma_u^2}+\frac{\tilde{v}^2}{\sigma_v^2}\right]\right)\cos\left(\frac{2\pi\tilde{u}}{\lambda}\right) \qquad (1)$$

where

$$\tilde{u} = u\cos\varphi + v\sin\varphi \quad \text{and} \quad \tilde{v} = -u\sin\varphi + v\cos\varphi.$$

The parameters φ and λ determine the orientation and period of the sinusoidal plane wave, respectively, while σ_u and σ_v control the standard deviations of the Gaussian envelope.

We use a bank of 32 identical rotated 7×7 pixel Gabor filters given by (1) with $\sigma_u = \sigma_v = 1$, $\lambda = 3$ and orientations ranging from 0° to 180°. Figure 5 shows sample plots of four such filters. Each photograph is convolved with the whole filter bank and the angle of maximum response is recorded for every pixel in the image.

In the traditional Canny filter, non-maximum suppression defines a pixel to be part of an edge if the gradient magnitude assumes a maximum in the direction normal to the local edge orientation. To create a ridge detector instead, we simply look for a maximum in the blurred input image itself: for a pixel with intensity M, let M_1 and M_2 be the bilinearly interpolated values resulting from a lookup normal to the local ridge orientation obtained from the Gabor filter. We create a new image with intensities given by

$$M' := \frac{M - M_{\max}}{M_{\max} + c}$$

where $M_{\max} := \max\{M_1, M_2\}$ and c is a constant. Hysteresis thresholding is then applied to this image and the resulting sparse feature information is retained. This relative peak finder helps to detect both bright features and hairs receiving very little light due to shadows. A purely relative threshold will generally suffer from spurious detections caused by noise in dark regions of the image, and we therefore adjust the the parameter c to reduce this to a minimum. To avoid artifacts resulting from camera demosaicing and

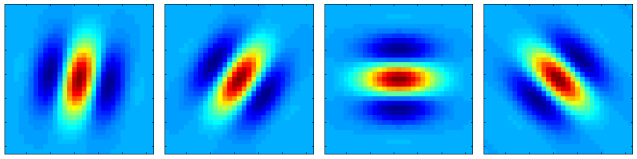

Figure 5: *To detect the image-space orientation of in-focus hairs, each photograph is convolved with a large filter bank of rotated Gabor filters. The angle of maximum response serves as a estimate of the projected fiber direction.*

range reduction, the filter is applied to the raw Bayer grid values corresponding to only one of the color channels.

The significant sparseness introduced by the filter is essential to later stages of the procesing pipeline. By storing only the positions and ridge orientations of pixels on focused hairs, the entire dataset of 24 sweeps can from now on be kept in memory.

4.2 Fiber-growing algorithm

Reliably finding hairs in the resulting data is challenging for several reasons: Despite the filter's resilience to noise, a certain number of spurious detections necessitate a robust method that is not unduly distracted by contradictory information. Secondly, many hairs are in contact, and the method has to handle such cases without generating curves with incorrect topology at the intersection points. Finally, the significant overlap of ribbons (Figure 3) from hairs in close proximity could be mistaken for hairs where none are to be found in the original.

We propose a fiber-growing algorithm that traces hair fibers using an alternating sequence of least squares iterations. As discussed in section 3, the path of a hair lies at the common intersection of a set of intersecting ribbons, each of which corresponds to a particular observation angle. Given an initial seed point on a hair, the algorithm estimates the local direction and takes an according step. Afterwards, a correction is applied to ensure that the new position still lies at the intersection of the ribbons (Figure 6). Each of these steps can be formulated as a least squares problem and the process is repeated on both ends, growing the hair outwards until the ends are reached or the hair passes out of sight.

The algorithm relies on the ability to query the set of detected features in the shared coordinate system described in Section 2. The particular type of search query used will be to look for observations contained inside an oriented box. This operation can efficiently be implemented by storing the feature data inside a kd-tree.

Direction estimation: Given a point $\mathbf{x} = [x, y, z]^T$ on a fiber focused from m different camera angles $\theta_{i_1}, \ldots, \theta_{i_m}$ with image-space positions $\mathbf{u}_1, \ldots, \mathbf{u}_m$, translations $d_1, \ldots, d_m$, and 2D feature orientations $\mathbf{o}_1, \ldots, \mathbf{o}_m$, we can use the available information to deduce the local fiber direction near $\mathbf{x}$. Suppose that the distance from the camera center to the focal plane is given by l. Using homogeneous coordinates and assuming an ideal pinhole camera model, we have $\mathbf{u}_k = P_{k,d_k}(\mathbf{x})$ where

$$P_{k,d}(\mathbf{x}) := \begin{bmatrix} \cos\theta_{i_k} & 0 & \sin\theta_{i_k} & 0 \\ 0 & 1 & 0 & 0 \\ -\frac{\sin\theta_{i_k}}{f} & 0 & \frac{\cos\theta_{i_k}}{f} & \frac{d+l}{f} \end{bmatrix} \mathbf{x} \qquad (2)$$

and f is the camera's focal length. The set of rays parameterized by $P^{-1}_{k,d_k}\{\mathbf{u}_k + \alpha\mathbf{o}_k\}$ ($\alpha \in \mathbf{R}$) sweeps out a plane containing both $\mathbf{x}$ and the camera at rotation angle θ_{i_k}, and furthermore projects onto the view plane as a line with orientation $\mathbf{o}_k$. Let $\mathbf{n}_k$ denote the normal of this plane. In estimating the local hair direction, then, we are interested in a direction that simultaneously lies in the planes corresponding to each of the observation angles. However, due to measurement errors, we can generally only hope to find an approximate answer. The least squares solution to this problem is readily found using the singular value decomposition and the minimizer $\mathbf{g}$ is given by the m-th right singular vector of $[\mathbf{n}_1, \ldots, \mathbf{n}_m]^T$ where the $\mathbf{n}_i$ are column vectors. The predicted next position is then given by $\tilde{\mathbf{p}}^{k+1} := \mathbf{p}^k + h\mathbf{g}$, where h is the step size.

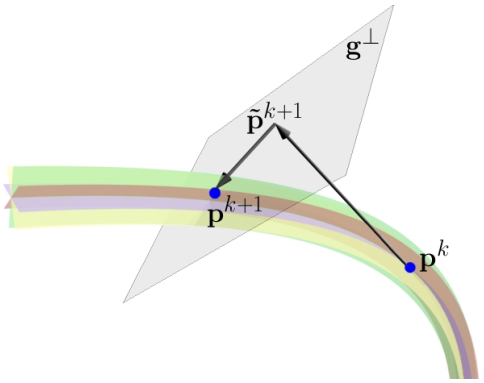

Figure 6: *Our algorithm iteratively moves along a hair identified by the common intersection of a set of mutually intersecting ribbons. This is done by repeatedly estimating the local fiber direction and taking a step, after which a correction moves the position back onto the centerline.*

Re-centering correction: Based on this minimization, a simple algorithm might move along a hair by repeatedly solving for a direction, taking a step, and then acquiring updated orientation information. The shortcoming of any such approach is that inaccuracies cause the iteration to drift away from the centerline, at which point the algorithm cannot reliably solve for a direction anymore. We require a re-centering mechanism to prevent this from happening.

To move back onto the centerline, we apply a correction perpendicular to the current growth direction (Figure 6). To compute the appropriate offset, the algorithm starts by searching for hair observations near the new position. This search is facilitated by introducing additional knowledge about the shape of ribbons near a point $\mathbf{x}$.

When $\mathbf{x}$ is positioned on camera k's focal plane, a small movement along the translation axis induces a corresponding shift to the projection of $\mathbf{x}$ in image space. In the camera's local (u, v, d) coordinate system, this shift can also be interpreted as a vector $\tilde{\mathbf{s}}_k := [\Delta u_k/\Delta d, \Delta v_k/\Delta d, \Delta d]$ describing the slope of a chain of feature detections extending through several images as the camera translates back and forth. When rotated into the shared coordinate system, this vector, together with the local hair direction, spans a plane that locally contains the ribbon observed by the camera. It is near this plane that we will search for hair observations. The vector $\tilde{\mathbf{s}}_k$ is found through differentiation of the camera transformation with respect to d and is given by:

$$\tilde{\mathbf{s}}_k = \begin{bmatrix} -f(x\cos\theta_{i_k} + z\sin\theta_{i_k})/l^2 \\ -fy/l^2 \\ 1 \end{bmatrix}$$

After rotating $\tilde{\mathbf{s}}_k$ into the shared coordinate system and constructing an orthonormal basis $\{\mathbf{g}, \mathbf{s}_k\}$ of $\mathrm{span}\{\mathbf{g}, \tilde{\mathbf{s}}_k\}$, we have obtained an orthonormal basis to the plane approximation, which we then complete to a full basis $\{\mathbf{g}, \mathbf{s}_k, \mathbf{t}_k\}$ of $\mathbf{R}^3$. We choose these as the axes of an oriented box with extents $\{a_g, a_s, a_t\}$ centered around $\mathbf{x}$, which can be used to query for hair observations.

```
GROW-HAIR()
 1  (p^0, g^0) ← GENERATE-SEED()
 2  k ← 0
 3  repeat
 4     (o_1, ..., o_m) ← FIND-ORIENTATIONS(p^k, g^k)
 5     if m < 3
 6           break
 7     g^{k+1} ← SOLVE-DIRECTION(p^k, o_1, ..., o_m)
 8     p̃^{k+1} ← p^k + h g^{k+1}
 9     p^{k+1} ← p̃^{k+1} + SOLVE-CORRECTION(p̃^{k+1}, g^{k+1})
10     k ← k + 1
11  REMOVE-FEATURES(p_0, ..., p_{k-1})
```

Figure 7: *By solving an alternating sequence of least squares systems, the fiber-growing algorithm iteratively steps along the centerline of a set of intersecting ribbons (Figure 6). The growing process stops once the number of observations drops below two, and the hair is subsequently removed to avoid future re-detection.*

The motivation of this approach is that, on the scale of single steps taken by the algorithm, a plane approximation is a good model for the shape of a ribbon, thus providing information on where feature detections are likely to be found. By restricting the search in this manner, large amounts of of detections from close-by hairs can be discarded, which is crucial to the stability of the fitting procedure described below. This search operation is executed with an appropriately chosen box for each camera angle (see Table 1 for parameters), and the resulting features are subsequently projected onto the two-dimensional subspace $\mathbf{g}^\perp$ orthogonal to the growth direction (Figure 8). Let $\mathbf{X}, \mathbf{Y}$ denote an orthonormal basis of this space.

When projecting the ribbon approximation $\mathrm{span}\{\mathbf{g}, \mathbf{s}_k\}$ onto $\mathbf{g}^\perp$, it, by definition, maps to a line of orientation $[\mathbf{X} \cdot \mathbf{s}_k, \mathbf{Y} \cdot \mathbf{s}_k]^T$. Given observations $\mathbf{x}_i$ seen by the associated camera, we therefore expect that, for an unknown offset $\mathbf{q} \in \mathbf{R}^2$, their projections will lie very close to this line (Figure 8), i.e:

$$\left([\mathbf{X}\ \mathbf{Y}]^T (\mathbf{x}_i - \tilde{\mathbf{p}}^{k+1}) - \mathbf{q} \right) \cdot \begin{bmatrix} -\mathbf{Y} \cdot \mathbf{s}_k \\ \mathbf{X} \cdot \mathbf{s}_k \end{bmatrix} \approx 0 \qquad (3)$$

Combining observations from several different camera angles, this results in a large least squares system, which can be solved for $\mathbf{q}$, e.g. using a QR factorization. The solution denotes the correction offset perpendicular to $\mathbf{g}$ and we then set $\mathbf{p}^{k+1} := \tilde{\mathbf{p}}^{k+1} + [\mathbf{X}\ \mathbf{Y}]\mathbf{q}$.

Finding 2D orientations: After a corrected step has been taken, the next iteration's direction fitting procedure requires updated 2D orientations at the new position. Another set of oriented box queries is executed around $\mathbf{p}^{k+1}$—this time with a much smaller value of a_s (e.g. 0.1mm), which results in observations where the hair was very close to the focal plane. The orientations of these results are then averaged separately for each camera angle.

Seed generation: Given a point on a hair, the proposed scheme can be used to track its path outwards in both directions. However, a method for finding the starting point is still required: here, we use a simple heuristic, which rasterizes the three-dimensional volume of observations into a grid and looks for the cell that records the highest number of observation angles. If this number is sufficiently large, we take this as evidence of a hair passing through the cell. To find its direction, the observed 2D orientations inside the cell are separately averaged for each camera angle, after which the SVD-based direction computation is applied. Finally, a full correction step is taken to let the starting position snap onto the centerline.

Rasterizing the entire sparse volume requires excessive amounts of memory, and we therefore only grow hairs from a smaller cube-shaped region that iteratively traverses the volume. Whenever the highest number of observation angles in any of its cells falls below a certain threshold (4 in our case), the cube advances by one position, and the process is stopped after the whole dataset was covered. This heuristic occasionally produces incorrect seeds, and we therefore also check that a hair can be grown for more than a specified minumum distance of a few centimeters. When this is the case, all observations used during the fitting process are deleted, thus preventing future re-detection of the same hair.

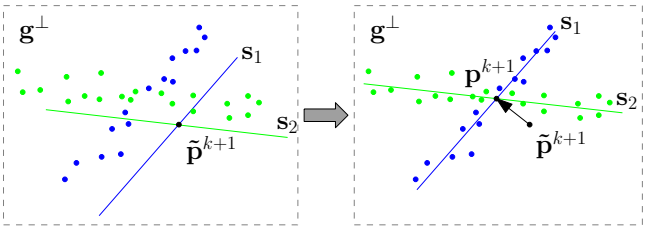

Figure 8: *A correction step adjusts the position in the plane perpendicular to the current growing direction. The offset is chosen to align hair observations with a model predicting their positions along a set of lines corresponding to the different camera angles.*

Enforcing smoothness: To reliably track a hair, we take very small steps, with $h = 0.1$mm. Implemented without further precautions, however, the algorithm frequently produces meandering curves that skip between hairs. This can happen when, despite the oriented box queries, significant amounts of noise and observations of other hairs creep into the described fitting procedures. A simple adaptation greatly increases the quality of the output: we make use of the comparatively low curvature of hair fibers by constraining the computed growth direction $\mathbf{g}^{i+1}$ to lie in small of cone of directions of size ω around the previous $\mathbf{g}^i$ and clamp if necessary. Similarly we enforce the correction offset to be below an upper bound γ in magnitude. Our bounds listed in Table 1 are chosen to let hairs curve freely while damping noise in the fitting process. Another possibility for further reducing noise is to extend the oriented box queries to the angular domain: given a pair $(\mathbf{p}^k, \mathbf{g}^k)$, their 2D orientation on each camera is easily obtained, and we can also restrict the search to nearby features in this space. Finally, we allow the algorithm to briefly continue growing a hair as a straight line if there are less than three observations. In many cases, an interruption of the ribbons can be bridged this way.

4.3 Generating additional hair

When renderable geometry is desired, it is necessary to fill in regions that cannot be reconstructed due to occlusion. We first build a voxelized bounding volume of the hair assembly using two separate methods. Each suffers from certain weaknesses, and their results are then combined to arrive at a volume of improved quality.

Visual hull: All photographs are taken in front of a bluescreen, and this provides a convenient way of bounding the space occupied by the hair assembly. Whenever the color of a pixel in one of the input images contains a sufficient fraction of blue relative to the other channels, we take this as evidence that a small region around the corresponding point on the focal plane contains no hair. Note that, because of the very shallow depth of field, we make no statement in terms of camera rays, since the assumption of emptyness does not necessarily hold true at a distance to the focal plane. After iterating through all images and keeping track of the highest observed amount of relative blueness in a voxel grid, we obtain a hull whose isosurface is shown in Figure 9. The weakness of this approach is that it cannot capture concave regions of the hair assembly.

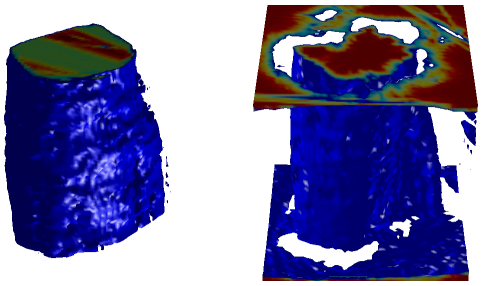

Figure 9: *The visual hull obtained using the bluescreen (left) is useful as a rough bounding shape, but fails to capture geometric depressions. The space carving volume is more accurate in this regard, but suffers from many areas incorrectly labeled as occupied. We combine the two volumes to remove both of these problems.*

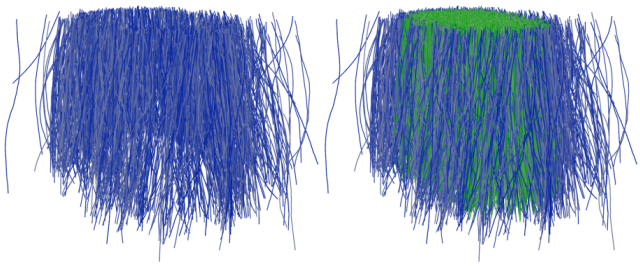

Figure 10: *Reconstructed fibers for the straight hair dataset. The right image also shows the generated in-fill hairs.*

Space carving volume: Here, feature detections from the filter stage are taken as evidence that there is little occlusion on the ray segment connecting them to the camera. All features, along with camera positions from which they were observed, are transformed into a shared coordinate system and used to trace rays through a voxel grid, keeping track of the total number of rays passing through any cell. An isosurface of the ray counts is shown in Figure 9. This approach resolves concave parts of the hair assembly, but fails to cull away many regions of empty space that are not intersected by any rays. To arrive at bounding volume of better overall quality, we use the visual hull as a stencil to remove incorrectly classified regions from the space carving volume.

Hair generation: Using an advection scheme, we populate occluded regions with artificially generated hair. First, average hair directions are voxelized and extrapolated into empty regions of the volume. This is done by a component-wise Laplace equation-solve for the directions, where non-empty cells specify the boundary conditions. To arrive at a smooth transition between reconstructed and generated hair, we blur a thresholded version of the previously obtained bounding volume. Finally, we sample positions according to these fractional densities and use the midpoint method to advect the ends of a new hair strand through the extrapolated velocity field. Growth is stopped once the fiber exits the scene bounds, and we then decide whether or not to accept this hair. A good criterion is the density integrated along the fiber and relative to its length. If this value is very low, the hair has traveled too far outside of the core, and is therefore removed. This basic advection scheme generally produces satisfactory results on simple hairstyles. For other cases, the method presented by Paris et al. [2008] may be more suitable.

5 Results

Our data processing pipeline, implemented in C++ and Matlab, ran on a 4-core Intel Core i7 940 workstation with 6 GB of memory. Processing one capture, which consisted of approximately 100GB of image data in the camera's proprietary raw format, entails first converting all the images to standard formats, running the C++ feature detection and fiber growing pipeline, and running the Matlab infill hair generation pipeline. Our unoptimized pipelines ran in about 12 hours, with equal time spent in the filtering and fiber growing stages (visual hull, space carving, and infill fiber generation take negligible additional time).

We used three switches of human hair, each roughly 25 cm in overall length. The first sample, *straight*, was medium brown hair, combed straight. The second sample, *wavy*, was blond hair, curled lightly with a curling iron to introduce gentle waves. The third sample, *curls*, was dark brown hair, curled more tightly to produce a complex arrangement of ringlets. In addition to the 9024 measurement images, we also acquired a turntable sequence of 360 well focused images using a small aperture, for validation.

The final models were rendered using a method based on the work of Moon et al. [2008] and the scattering model parameters were chosen to match the photographs. Multiple scattering was only included for the blond hair sample. Rendering a single frame took approximately 20 minutes, on an 8-core Intel Xeon system.

For the *straight* hairstyle, the reconstructed curves are shown in Figure 10, both alone and together with the infill fibers. For this hairstyle we extract a total of 2443 curves, with an average of 1213 curves crossing any given horizontal plane. Near the outside of the hair, most fibers are complete from top to bottom, and as the hairs get more occluded towards the inside, we find smaller segments. To illustrate the fraction of visible fibers that are captured, in Figure 14 we overlay the projections of the recovered curves on the measurement images. The vast majority of fibers that are clearly focused result in fibers in the model. We added 4000 infill fibers to this data, and the rendering is shown compared to a turntable image in Figure 12. (See the supplementary video for the full feature-overlay sequence and the full turntable comparisons.)

The *wavy* hairstyle contains many well-aligned fibers that pack more tightly together, combined with flyaway hairs, and the overall structure allows some views into the interior. The method continues to perform well under these conditions, and we extract a total of 1527 curves, with an average of 789 curves crossing any given horizontal plane, and produce 4000 additional infill fibers. The turntable comparison is shown in Figure 13.

In the *curls* hairstyle, our method reproduces most of the flyaway hairs, but encounters difficulties in the dense, well-aligned regions. As the surface bears increasing resemblance of that of a solid object, individual fibers become harder to track, as they cannot be clearly resolved separately from one another. Here, approximate methods

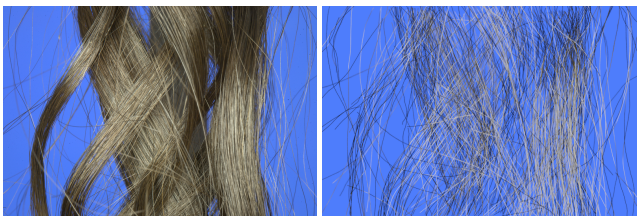

Figure 11: *In this hairstyle, the high amount of coherence and resulting dense packing presents a difficult case for our algorithm (Left, photograph; right, reconstruction without infill fibers).*

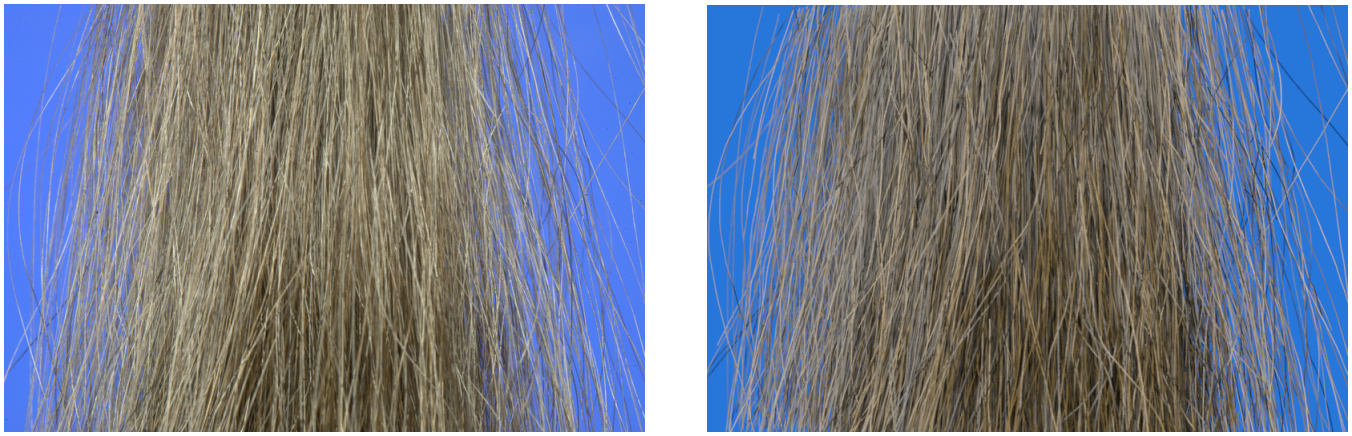

Figure 12: ***Straight hairstyle****: Comparison of the rendered hair geometry obtained using our approach (right) against a reference photograph shown on the left. Note the accurate reproduction of many image features including flyaway hairs.*

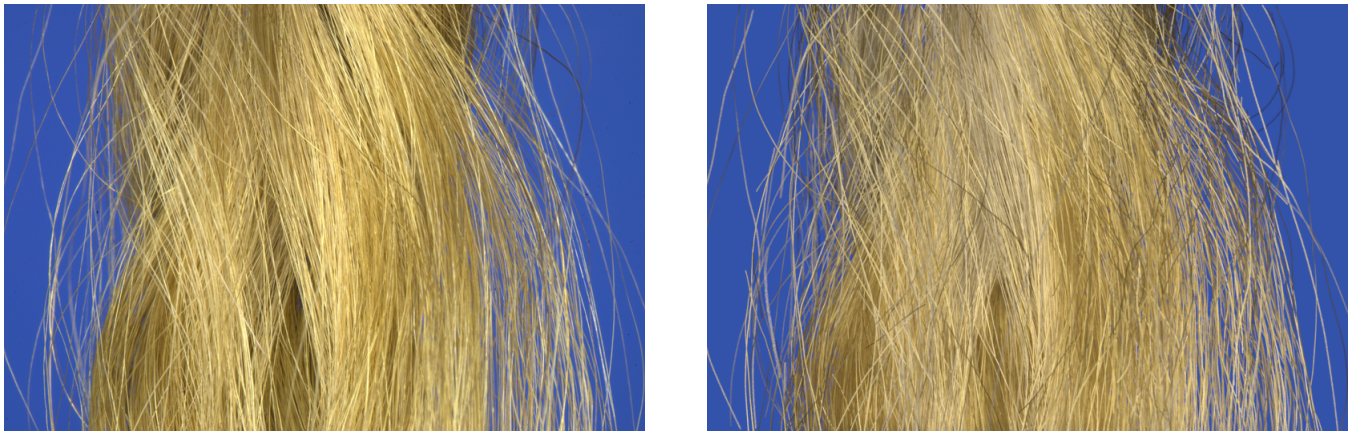

Figure 13: ***Wavy hairstyle****: Rendered curves (right) compared against a reference photograph (left). With the high-quality reconstruction, lighting matched to the capture setup, and a physically-based renderer, we can produce highly realistic results. See the accompanying video for a more complete comparison.*

	c	ω	γ	a_g	a_s	a_t	# Fibers
Straight	-140	1.5°	.5px	40	30	8	1213
Wavy	-90	1.5°	.5px	60	30	8	789
Curls	-70	1.5°	.5px	60	30	8	404

Table 1: ***Parameters:*** *Filter parameters, smoothness constraints, and box query sizes in pixels for each of the datasets, along with the average number of reconstructed fibers per horizontal slice.*

such as [Wei et al. 2005] may be better-suited, since they generate plausible geometry in such regions of high density.

6 Conclusions

Whereas previous work in hair capture has always aimed only to capture occupancy and a map of local fiber orientations, we have demonstrated the feasibility of directly capturing the geometry of a large fraction of the visible fibers in a hairstyle. In contrast to previous work that randomly generates local structure, our models have realistic 3D arrangements of fibers, providing the first looks at the actual geometric behavior of fibers in real hairstyles.

Of course, the small field of view limits the direct practical application of our method to capturing hairstyles. Larger and higher resolution cameras, or many tiled acquisitions using the same type of camera, would enable our method to be used directly to capture larger hairstyles. Alternatively, higher quality local structure might be to use our system to capture examples of local arrangements, and then use those examples to generate realistic fibers that fit into the global shape established by existing modeling tools.

The ability to capture fully detailed hair models has significant implications for the development and evaluation of methods for rendering and simulating hairs. It enables comparison to ground truth images and geometry, which will reveal the limitations of rendering and simulation methods and spur their further development.

7 Acknowledgements

This work was supported by the National Science Foundation (grants CCF-0347303 and CCF-0541105) and by Unilever Corporation. Computing equipment was generously provided by Intel. The authors also thank Noah Snavely and Robert Velthuizen for valuable conversations that helped shape this work.

(a) *Input image* **(b)** *Gabor filter response* **(c)** *Projected curves*

Figure 14: *To validate our results, we project the recovered curves back onto the input images, discarding segments that are too far away from the focal plane. The close resemblance to the Gabor filter response shows that most image features result in output curves.*

References

ANJYO, K.-I., USAMI, Y., AND KURIHARA, T. 1992. A simple method for extracting the natural beauty of hair. In *Proceedings of SIGGRAPH '92*, 111–120.

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 *Symposium on Computer Animation, SCA 2003*, 207–213.

BERTAILS, F., AUDOLY, B., CANI, M.-P., QUERLEUX, B., LEROY, F., AND LÉVQUE, J.-L. 2006. Super-helices for predicting the dynamics of natural hair. *ACM Transactions on Graphics 25*, 3 (July), 1180–1187.

CANNY, F. J. 1986. A Computational Approach to Edge Detection. *IEEE Trans. Pattern Anal. Mach. Intell. 8*, 6, 679–698.

CHOE, B., AND KO, H.-S. 2005. A statistical wisp model and pseudophysical approaches for interactive hairstyle generation. *IEEE Trans. Vis. Computer Graph. 11*, 2 (Mar./Apr.), 160–170.

CURLESS, B., AND LEVOY, M. 1996. A volumetric method for building complex models from range images. In *Proceedings of SIGGRAPH 96*, 303–312.

ENS, J., AND LAWRENCE, P. 1993. An investigation of methods for determining depth from focus. *PAMI 15*, 2 (Feb), 97 – 108.

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.

HADAP, S., AND MAGNENAT-THALMANN, N. 2000. Interactive hair styler based on fluid flow. In *Computer Animation and Simulation 2000*, 87–99.

HASINOFF, S. W., AND KUTULAKOS, K. N. 2009. Confocal stereo. *Int. J. Comput. Vision 81*, 1, 82–104.

INOUÉ, S., AND SPRING, K. R. 1997. *Video Microscopy: The Fundamentals*, second ed. Plenum Press, New York.

JAIN, A. K., AND FARROKHNIA, F. 1991. Unsupervised texture segmentation using gabor filters. *Patt. Rec. 24*, 12, 1167–1186.

JONES, J. P., AND PALMER, L. A. 1987. An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex. *J. of Neurophysiol. 58*, 6 (December), 1233–1258.

KIM, T.-Y., AND NEUMANN, U. 2000. A thin shell volume for modeling human hair. In *Computer Animation 2000*, 104–111.

KIM, T.-Y., AND NEUMANN, U. 2002. Interactive multiresolution hair modeling and editing. *ACM Trans. Graph. 21*, 3, 620–629.

LEVOY, M., NG, R., ADAMS, A., FOOTER, M., AND HOROWITZ, M. 2006. Light field microscopy. *ACM Transactions on Graphics 25*, 3 (July), 924–934.

LINDEBERG, T. 1998. Feature Detection with Automatic Scale Selection. *Int. J. Comp. Vis. 30*, 2 (November), 79–116.

MOON, J. T., AND MARSCHNER, S. R. 2006. Simulating multiple scattering in hair using a photon mapping approach. *Proceedings of SIGGRAPH 2006 25*, 3, 1067–1074.

MOON, J. T., WALTER, B., AND MARSCHNER, S. 2008. Efficient multiple scattering in hair using spherical harmonics. In *Proceedings of SIGGRAPH 2008*, vol. 27, 31.

PARIS, S., BRICE NO, H. M., AND SILLION, F. X. 2004. Capture of hair geometry from multiple images. *ACM Trans. Graph. 23*, 3, 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. In *Proceedings of SIGGRAPH 2008*, 30.

SELLE, A., LENTINE, M., AND FEDKIW, R. 2008. A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3 (Aug.), 64.

SOBOTTKA, G., KUSAK, M., AND WEBER, A. 2006. Hairstyle Construction from Raw Surface Data. In *Proceedings of CGIV 2006*, IEEE Computer Society, 365–371.

WANG, L., YU, Y., ZHOU, K., AND GUO, B. 2009. Example-based hair geometry synthesis. In *Proceedings of SIGGRAPH 2009*, ACM, New York, NY, USA, 1–9.

WARD, K., BERTAILS, F., KIM, T.-Y., MARSCHNER, S. R., CANI, M.-P., AND LIN, M. 2007. A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics (TVCG) 13*, 2, 213–34.

WEI, Y., OFEK, E., QUAN, L., AND SHUM, H.-Y. 2005. Modeling hair from multiple views. *ACM Trans.Graph. 24*, 3, 816–820.

YANG, J., LIU, L., JIANG, T., AND FAN, Y. 2003. A modified Gabor filter design method for fingerprint image enhancement. *Pattern Recognition Letters 24*, 12 (August), 1805–1817.

YU, Y. 2001. Modeling realistic virtual hairstyles. In *9th Pacific Conference on Computer Graphics and Applications*, 295–304.

ZINKE, A., AND WEBER, A. 2007. Light scattering from filaments. *IEEE Transactions on Visualization and Computer Graphics 13*, 2, 342–356.

ZINKE, A., YUKSEL, C., WEBER, A., AND KEYSER, J. 2008. Dual scattering approximation for fast multiple scattering in hair. *ACM Transactions on Graphics 27*, 3 (Aug.), 32.