Pacific Graphics 2012
C. Bregler, P. Sander, and M. Wimmer
(Guest Editors)

Volume 31 (2012), Number 7

Simulation Guided Hair Dynamics Modeling from Video

Qing Zhang[1], Jing Tong[2], Huamin Wang[3], Zhigeng Pan[2] and Ruigang Yang[1]

[1]University of Kentucky, Lexington, KY, USA
[2]Zhejiang University, Hangzhou, Zhejiang, China
[3]Ohio State University, Columbus, OH, USA

Abstract

In this paper we present a hybrid approach to reconstruct hair dynamics from multi-view video sequences, captured under uncontrolled lighting conditions. The key of this method is a refinement approach that combines image-based reconstruction techniques with physically based hair simulation. Given an initially reconstructed sequence of hair fiber models, we develop a hair dynamics refinement system using particle-based simulation and incompressible fluid simulation. The system allows us to improve reconstructed hair fiber motions and complete missing fibers caused by occlusion or tracking failure. The refined space-time hair dynamics are consistent with video inputs and can be also used to generate novel hair animations of different hair styles. We validate this method through various real hair examples.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—Line and curve generation

1. Introduction

With fast advances in digital imaging sensors and computer vision algorithms, more objects and natural phenomena, including buildings, plants, human bodies and even liquid, can now be captured and reconstructed fully in 3D. The resulting 3D models can be rendered from new viewpoints, in different lighting conditions, or with other graphics models in various graphics and visualization applications. The ability to create 3D models with little or no user interaction alleviates one of the central problems in computer graphics: *modeling*.

Recent image-based hair reconstruction approaches [WOQyS05,PCK*08] show that realistic static 3D hair models can be recovered from regular images. However, extending their success to dynamic cases is challenging, because the requirement for high-resolution images or multiple frames under controlled lighting environment [PCK*08] is especially difficult to meet under fast hair motion. Even if one could build expensive high-speed hardware to reconstruct each frame individually, it is still difficult to maintain temporal consistency over time.

Our solution to this hair dynamics modeling problem is to use physically based simulation to improve the low-fidelity dynamic models. Using a set of synchronized multi-view video sequences as input, our dynamic hair modeling system first reconstructs individual hair fibers at each time instant. Because of the low-resolution images and uncontrolled lighting, the individual hair model may have multiple outliers and inconsistency errors over time. Therefore we develop a simulation-based refinement method to generate temporally consistent space-time hair model.

While the idea of combining image-based reconstruction with physically based simulation has been used to reconstruct fluid animation from videos (*e.g.*, [WLZ*09]), the objective in hair modeling is different. In the case of fluid, the underlying physics is well defined and the practical concerns are about efficiency, stability, and resolution, which are also research topics in simulation; in the case of hair, the complicated interactions among hair fibers are not well modeled or understood. Recent hair simulation methods are mainly based on two kinds of models: the mass-spring system [SLF08] and the super-helices model [BAC*06].

Inspired by the hybrid hair simulation method proposed by McAdams and his colleagues [MSW*09], we present our hybrid hair dynamics refinement algorithm using Lagrangian particle-based hair simulation and continuous Eulerian fluid simulation. Differently from the pure physical simulation, our refinement algorithm is formulated as a nonlinear optimization procedure. In particular, we use the particle-based simulation to guide hair dynamics and optimize them to be consistent with the initially reconstructed hair sequence as much as possible. The whole algorithm allows us to incorporate fine motions at fiber level, and overcome the problem of insufficient constraints for multiple occluded or un-trackable hair fibers.

Our approach is novel in the following ways: (a) since we use real-world reconstructed data as constraints, we assume fiber collisions have already been included so that we do not need to explicitly model collisions in simulation; (b) we find physical parameters of hair fibers automatically from the data by treating them as variables in the optimization process, making it more convenient to control the simulation compared to adjusting all the parameters manually; (c) we develop a simple method to generate novel hair animations of different hair styles using the captured motion.

Our experiments show that the refinement approach can smoothly recover hair dynamics from input videos, even the initial image-based reconstruction is noise or detail-missing in the temporal sequence. For complicated hair styles that are difficult to reconstruct directly from videos, our method provides an option for users to drive the static user-specific hair geometry model using the the reconstructed dynamics as shown in Figure 7. Since our system requires only regular multi-view video streams as input, we hope it can be used in concert with image-based face modeling or performance capture approaches. Eventually, the reconstructed human character model should be complemented with realistic and dynamic hair, as we typically see in the real world.

2. Related Work

[WBK*07] presented a survey on hair modeling approaches during the past decade, including both styling and simulation. A particular category of these approaches, image-based hair modeling, can be used to automatically generate realistic hairstyles. Recent methods include the multiple-view approach [WOQyS05] and the hair photobooth [PCK*08]. Wei *et al.* [WOQyS05] propose a multiple-view method to triangulate multiple 2D orientation maps into 3D orientation vectors for visible hair, and then synthesize hair strands via measured orientations from roots. This approach is automatic with little user interaction except for hair masks. The problem is that it can only handle visible hair regions in the image and the result looks like a thin shell. Extended from the multi-view approach, the hair photobooth method [PCK*08] models the 3D orientation as a 3D vector field and then utilizes structure tensors to complete a dense orientation field. A dedicated acquisition setup is used to measure the orientation on the hair surface. [LLW*11, LLP*12] utilize the local orientation as a robust matching criterion to recover the depthmap of hair surface among multiple views and grow hair in the refined 3D orientation field in a similar way of [PCK*08] for each frame. [YWO09] uses Wei's approach and applies a temporal smoothing to generate a hair sequence. Since their smoothing operator over-constrains the growth of hair fibers, they can only model a uniform motion of straight hair on a stationary head.

Early physically based hair simulation approaches include the original mass-spring system [RCT91] and the projective dyanmics method [iAUK92]. These approaches typically treat each individual hair strand as a string of particles with sufficient degrees of freedom in translation and rotation and they are not able to handle hair twist. Later, [HMT01] introduces the rigid body chain and models hair contact using viscosity. [CCK05] simulates rigid body links using guide hairs with a statistical model. [Had06] uses the articulated rigid body chain to model hair and foliage. Although rigid body schemes can model the twist and enforce hair length preservation, they are usually implicitly involved and difficult to constrain.

To represent the strand dynamics more accurately, mechanical models have been adopted into the graphics community. [Pai02] introduces the Cosserat model for flexible rods by representing hair strands using the Kirchhoff model. [ST07] discretizes the hair motion in the quaternion space and simulated knots in real time. Bertails *et al.* [BAQ*05] present an energy minimization method to compute equilibrium strand positions under external forces. They later propose the super-helices model [BAC*06] to handle strand dynamics. In this model, each strand is represented by a sequence of helices, where the torsion and curvature are used as general coordinates to control the helix centerline. Furthermore, [Ber09] provides a linear method that reduces the computational cost of [BAQ*05] by recursively computing the inertia matrices from tip to root in an initial pass. Recently, [DBDB11] uses the super-helix model to solve the dry friction in hair dynamics.

Another trend is to improve the mass-spring system, [MBTF03] introduces an altitude spring model that can prevent volume collapse and make hair strands tend to recover to original shapes. Combining the altitude spring with the mass-spring system, [SLF08] and [SSIF07] demonstrate the efficiency to generate physically plausible hair dynamics. We prefer to the improved mass-spring method since it is more convenient to integrate in our optimization procedure, although the helices models can have more power in simulating complex hair styles.

3. System Overview

The objective of our hair modeling system is to generate photo-realistic hair sequence from multiple-view video sequences. Figure 2 shows the overall reconstruction pipeline. We firstly use the existing image-based hair reconstruction algorithm [WOQyS05] to recover the hair geometry for each frame. The reconstructed hair is refined in the temporal refinement stage, where we use a particle-based simulator to refine the velocity of each hair fiber and then build the velocity field using the volumetric diffusion method. Eventu-

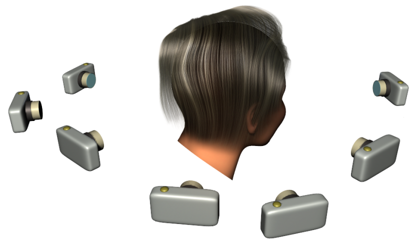


Figure 1: *The capture system setup consists of seven synchronized and calibrated DragonFly Express cameras at 100 FPS.*

ally, we can use the velocity field to polish hair geometry or generate new hair animation.

Our acquisition system consists of seven synchronized high-speed Dragonfly Express cameras of a 640 × 480 resolution at the 100 fps. They are placed approximately over a semi-circle and pre-calibrated as illustrated in Figure 1. In the data capture step, we place small markers on the top of head to help track the global head pose.

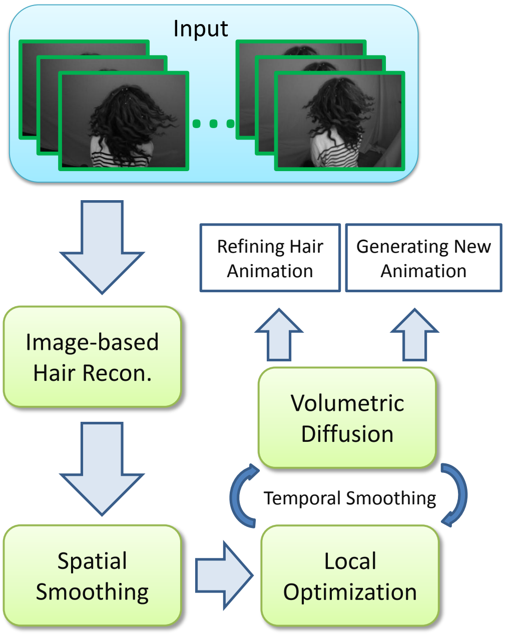


Figure 2: *The system pipeline of modeling hair dynamics.*

4. Initial Reconstruction

With a clean background, we firstly build the visual hull for each input frame and define a volume in 3D containing the whole visual hull sequence. In our experiment, we use a regular volumetric grid of 2mm voxel edge length. The hair volume for each frame is further defined as the 3D region between two shells: the hair outer surface (S_{hair}), which is a part of the visual hull, and the scalp surface (S_{scalp}), which contains the head scalp and hair roots like definitions in [WOQyS05].

Since S_{scalp} is not directly observable, we simply assume that S_{scalp} is a shrunk version of S_{hair}. In order to get a better estimate of S_{scalp}, we let the subject move slowly at the beginning of the acquisition and define the reference coordinate system using the initial head pose, which can be tracked from markers. The idea of using more images is to obtain a tighter visual hull estimate of S_{hair} (and S_{scalp}). Once we find S_{scalp}, the shape is fixed in the entire sequence and only transformed according to the global head pose, although S_{hair} can change over time.

To track the head pose, we stick markers on the top of the head and manually label them at the first frame of the video sequence, and then we track markers using optic flow algorithms [BA93] in each view. The marker's 3D position is them triangulated from multiple images. Let $\mathbf{m}_i^t$ be marker i's position at frame t, we estimate the head motion $\mathbf{C}^t$ from the initial pose to the frame t as follows:

$$\mathbf{C}^t = \arg\min \sum_{\mathbf{m}_i^t \in \mathbf{M}^t} \left\| \mathbf{m}_i^t - \mathbf{C}^t \mathbf{m}_i^0 \right\|^2, \qquad (1)$$

in which $\mathbf{M}^t$ is the set of successfully tracked markers in all views at frame t. In our experiment, the set $\mathbf{M}^t$ contains at least five successfully tracked markers during the entire sequence, and the average projection error of markers in each camera view is less than 5 pixels. For each frame, we transform the initial scalp S_{scalp} with $\mathbf{C}^t$ to get the scalp $S^t = \mathbf{C}^t S_{\text{scalp}}$.

Hair Fiber Synthesis We uniformly sample hair roots over the scalp S_{scalp}, which is tracked according to the head pose, and then we have temporal correspondences for all hair roots. But we do not impose temporal constraints during the hair growing process like the way in [YWO09] by the consideration that temporal correspondences among fibers are not reliable when the hair motion is large. Instead, we construct the initial hair fibers frame by frame using the basic strategy in [WOQyS05] and apply further temporal refinement in the next stage.

In the hair fiber synthesis, 2D orientations are firstly obtained using the gradient filter [WOQyS05], and then an undirected 3D line at a given 3D position can be triangulated from the 2D orientation of the projected location in multiple views. In our setup, a valid 3D line is reconstructed when at least three views are available and then assigned to a voxel. Typically, about 10 percent of the voxels in the hair volume can be faithfully assigned with a 3D orientation. To make the field dense, we use the hole filling process introduced by [PCK*08] to diffuse the volume. The idea is to use a 3 × 3 symmetric tensor matrix to encode each undirected 3D line, which allows tensor linear interpolations, and then blank voxels can be influenced by their neighbors after sufficient number of iterations.

With a complete 3D orientation field, we grow each hair fiber from its root and terminate when a predefined maximum length is reached, or when the fiber reaches the outer boundary of hair volume. Since the 3D orientation is undirected, the real direction is picked to be the one having a smaller angle to the previous direction. More implementation details can be found in [PCK*08].

After the fiber synthesis, we suppress outliers in two ways: the first is to project the reconstructed hair fiber back to images and then validate it according to 2D orientations; the second is to define a vector field and use an iteratively weight-changing scheme in [WYZG09], which enforces the direction towards the most common one in each voxel and improves the consistency among adjacent fibers.

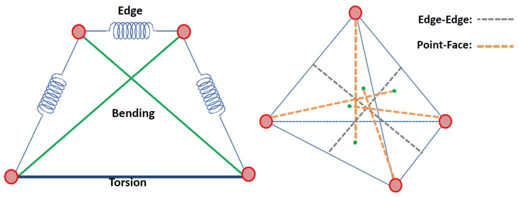
Spatial Smoothing Before we use the synthesized hair fiber as input to our temporal refinement stage, we perform a spatial smoothing to all the fibers, because the physically based simulation can easily fail due to the data noise accumulation and the synthesized fiber does not guarantee enough smoothness even though we grow each fiber segment at the smoother direction. Given the synthesized hair fiber $\tilde{\mathbf{x}}^t$ at time t, we uniformly sample the fiber by a certain edge length (5mm in our experiment) and denotes the n_t particles $\{\tilde{\mathbf{x}}_i^t\}, i = 1, \ldots, n_t$ as anchor points. The smoothing process on each fiber is to minimize the following energy functional:

$$\min_{\mathbf{x}^t}(1-w_s)\sum_{i=1}^{n_t}\left\|\mathbf{x}_i^t-\tilde{\mathbf{x}}_i^t\right\|^2+w_s\sum_{i=2}^{n_t}\left\|\left(\mathbf{x}_i^t-\mathbf{x}_{i-1}^t\right)-\left(\tilde{\mathbf{x}}_i^t-\tilde{\mathbf{x}}_{i-1}^t\right)\right\|^2, \tag{2}$$

in which w_s is a smoothing weight balancing the data term and smoothness term and we choose $w_s = 0.5$ in our experiment. The least square minimization is equivalent to a linear equation and hence we can solve new positions $\{\mathbf{x}_i^t\}$ of all points. The total effect is to bend the fiber smoothly from root to tip while maintaining the original shape. After smoothing all the fibers, we trim a fiber in the sequence (*i.e.*, $\{\mathbf{x}^t\}$, $t = 1, \ldots, n_t$) to a fixed number of points and take the result fiber sequence as the input to our temporal optimization algorithm in the following section.

5. Hair Dynamics Refinement

In this section, we introduce the hair dynamic refinement scheme using hair simulation, which deals with the evolution of states: positions $\mathbf{x}$, velocities $\mathbf{v}$ and accelerations $\mathbf{a}$ from time t to $t+1$. The implicit time integration [BW98] is preferred to handle highly stiff spring systems. In our temporal refinement scheme, we firstly use a *mass-spring* system [SLF08] to individually match the fiber motion observed from the input data and then generate the velocity field in the whole volume using a *fluid-dynamics* method. We note that the mass-spring simulation only deals with the behavior of individual fiber for efficiency and the fluid simulator mimics effects such as air friction and hair collisions. After the refinement, we will get a continuous velocity field in temporal sequence, which allows one to combine with other hair styles to synthesize new animation.



(a) A mass-spring configuration (b) Altitude springs

Figure 3: *The tetrahedron in (a) has edges, bending and torsion springs. Point-face and edge-edge altitude springs are placed in (b). In total, there are 4 pairs of point-face and 3 pairs of edge-edge springs.*

5.1. Mass Spring Model

We develop our simulation based on the mass-spring model for two considerations: (a) mass-spring systems formulate the relationship of each state both explicitly and linearly, thus easily to combine with various external constraints; (b) complicated physical models such as elastic rods are not necessary since we have already obtained guide information from videos and do not need to generate a pure physically based simulation. For the purpose of completeness, we briefly describe the kinematics of a mass-spring system and how to solve its dynamics. The reader can refer to [SLF08] for a detailed configuration.

We formulate each hair fiber as n mass points, where each particle is driven by the gravity and internal forces generated by four kinds of damped springs. An edge spring is placed at every edge to preserve the edge length. A bending spring connects between every other particles to present the bending ability. A torsion spring that connects each particle to a particle three particles away from it models the twist effect (see Figure 3a). To prevent volume collapse, seven altitude springs are place in every tetrahedron of four consecutive particles as shown in Figure 3b. The force of each altitude spring contributes to each vertex of the tetrahedron via the barycentric coordinates of the endpoint.

The total internal force on each particle is the sum of elastic forces from these four categories of damped springs. Since we have already known the head motion during the data capture, we simply ignore the external force on fiber roots and transform the fiber root using the head motion. During our simulation, the gravity is always constant towards a pre-calibrated direction. In addition, a viscous drag force, which is opposite and proportional to the velocity, is added to each particle to approximate the air friction.

5.2. Local Particle-based Optimization

The purpose of the local optimization is to move forward a certain fiber in a local temporal window as similar as the reconstructed data. The goal is two-fold: (a) to find the proper parameters that drive the correct motion, and (b) to refine the simulate geometry according to the reconstructed fibers. We use the stacked column vectors to represent positions and velocities of n particles in a fiber: $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^T$, $\mathbf{v} = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n]^T$. Suppose we choose a temporal window of size s and have $s+1$ reconstructed fiber positions: $\tilde{\mathbf{x}}^0, \tilde{\mathbf{x}}^1, \ldots, \tilde{\mathbf{x}}^s$. Picking an initial fiber $\mathbf{x}^0 = \tilde{\mathbf{x}}^0$, $\mathbf{v}^0 = \tilde{\mathbf{x}}^1 - \tilde{\mathbf{x}}^0$ and choosing an initial guess of stiffness parameters, we can perform the particle simulation in a Back-Euler scheme with a fully implicit integrator to get $s+1$ simulated fiber positions $\mathbf{x}^0, \mathbf{x}^1, \ldots, \mathbf{x}^s$:

$$\left\{ \begin{array}{l} \left(\mathbf{I} - \frac{\Delta t}{m}\frac{\partial \mathbf{f}}{\partial \mathbf{v}}\Big|_t - \frac{\Delta t^2}{m}\frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_t\right)\Delta\mathbf{v} = \frac{\Delta t}{m}\left(\mathbf{f}^t + \Delta t \frac{\partial \mathbf{f}}{\partial \mathbf{x}}\Big|_t \mathbf{v}^t\right), \\ \mathbf{v}^{t+1} = \mathbf{v}^t + \Delta\mathbf{v}, \\ \mathbf{x}^{t+1} = \mathbf{x}^t + \Delta t \mathbf{v}^{t+1}, \end{array}\right. \tag{3}$$

where the m is the particle mass, which we set to a unit and Δt is the simulation time step, which is 0.002 second in our experiment. Since all the forces can be expressed by fiber state and the parameters of springs, our goal is to find these parameters by minimizing the difference between the reconstructed and simulated fibers:

$$\mathbf{p} = \arg\min \left\{ \phi := \sum_{i=1}^{s} w_i \left\| \mathbf{x}^i - \tilde{\mathbf{x}}^i \right\|^2 \right\}, \tag{4}$$

where w_i is a Gaussian aging factor and the fiber difference is computed between the stacked column vectors. The nonlinear optimization provides refined parameters and a sequence of simulated fibers, matching the reconstructed data. In the idea case, given the initial state and proper parameters, we shall be able to simulate the whole sequence of dynamics. But in real world the dynamics of hair is complex, not only affected by intrinsic forces but also from some external forces like wind, hair friction and collisions. The simplification of intrinsic parameters and external interaction would make the simulation inaccurate. For instance, collisions may act as the main force on a fiber and disobey our simulation 3. Furthermore, due to the error accumulation, it becomes harder to control the exact motion in a longer period. To make the optimization 4 converge, we choose a local temporal window of a small size 5 and spread the local optimizations among the whole sequence by moving the temporal window forward. Once completing the optimization for the whole sequence, we update each fiber position $\tilde{\mathbf{x}}^t$ to the average position of corresponding simulated fibers that are generated in all possible windows. Taking the updated positions as reconstructed positions, we repeat the process until satisfied results show up. The whole optimization pipeline is shown in Figure 4.

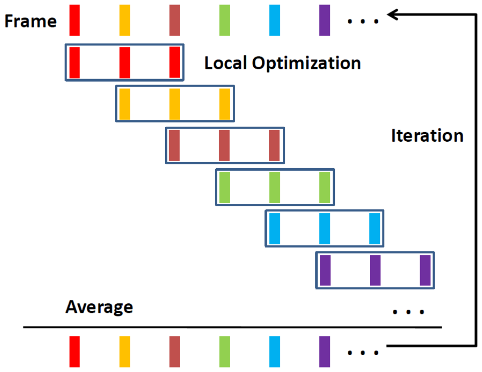


Figure 4: *The pipeline of our local optimization stage. Each local window contains a nonlinear optimization. Positions are updated as the average of fibers in all local windows. And then result positions act as the initial positions for the next iteration.*

One should note that each local optimization is still a nonlinear minimization problem with respect to spring parameters. Given the initial guess of user-specific parameters $\mathbf{p}_0$, we use the limited memory quasi-Newton optimization method (LBFGS) [ZBLN94] to solve this problem. The involvement of simulated frames and implicit method make it hard to compute the gradient $\partial\phi/\partial\mathbf{p}$ directly by finite differencing. We choose the adjoint method for gradient computation, which is widely used in simulating fluid [MTPS04] and cloth [WMT06] and turned out to be efficient to compute gradient implicitly. When the nonlinear optimization stops at a local minima, we does not guarantee to obtain a result matching exactly with the reconstructed data but the fiber motion is smooth in temporal as we show in the result section.

5.3. Volumetric Diffusion

The above simulation-based refinement scheme treats each individual hair fiber independently. In order to generate a continuous velocity field over space and time, which can be used to refine the existing hair motion 5.4 and generate new animation 5.5, we propose a diffusion step similar to the viscosity convection in fluid simulation. Using the hair volume grid defined in section 4, we set the initial value of each cell to the average velocity of all fiber passing through and zeros at elsewhere. Given a grid cell $\mathbf{x}$ and its velocity value $\mathbf{v}$, we update the velocity field on the hair volume grid as following:

$$\mathbf{v}(\mathbf{x}) = (1-6s)\mathbf{v}(\mathbf{x}) + \sum_{\mathbf{y}\in\mathbf{N}(\mathbf{x})} s\mathbf{v}(\mathbf{y}), \tag{5}$$

where s is a spatial smoothing kernel and specified according to the expectation of the velocity spatial smoothness, and the sum accounts for the grid cell's six immediate neighbors.

5.4. Refining Hair Animation

From the above velocity field, we can improve the quality of temporal coherence using interpolation. First, we define a

Hair Style	Fiber Count	Recon. Time /Frame	Total Opt. Time
Straight	7421	16min	1hr
Long Straight	7403	20min	1hr
Curly	7992	22min	2hrs

Table 1: *Performance of modeling various hair dynamics.*

Hair Style	Particle Count	Avg m/frame	total time
Textured Curly	240k	0.5min	1hr
Helix Curly	620k	1min	2hrs

Table 2: *Performance of simulating new hair animation using the captured velocity field.*

warped intermediate state $\widehat{\mathbf{x}} = \delta(\mathbf{x}, \mathbf{v})$, and the warping function δ is to solve the following minimization:

$$\sum_{i=1}^{n} s_i \left\| \widehat{\mathbf{x}}_i - (\mathbf{x}_i + \Delta t \mathbf{v}_i) \right\|^2 + s_l \sum_{i=1}^{n-1} \left(\left\| \widehat{\mathbf{x}}_{i+1} - \widehat{\mathbf{x}}_i \right\| - r \right)^2, \quad (6)$$

where r is the rest length constant (5mm in our case) and s_l is a common weight for each segment to preserve the length, s_i is set to the same for all vertices except for a large value s_1 to fix the root. We choose $s_l, s_i = 0.5$ in our experiment and set $s_1 = 100$ at the fiber root. The velocity $\mathbf{v}_i$ is the accessed at location of $\mathbf{x}_i$ in the velocity field.

Considering each fiber between two temporal neighbors, we then define two predicted intermediate positions $\mathbf{x}^- = \delta(\mathbf{x}^{t-1}, \mathbf{v}^{t-1})$ and $\mathbf{x}^+ = \delta(\mathbf{x}^{t+1}, -\mathbf{v}^{t+1})$, the optimized $\mathbf{x}$ is linearly interpolated using a relaxation coefficient $\gamma(\approx 0.25)$:

$$\mathbf{x} = (1 - 2\gamma)\mathbf{x} + \gamma(\mathbf{x}^+ + \mathbf{x}^-). \quad (7)$$

5.5. Generating New Animation

The velocity field sequence also allows us to generate new hair animations efficiently via hair simulation. Given a new hair style, we evolve the hair state according the captured velocity field, which may also be treated as temporal extrapolation of positions and velocities. First, we align hair roots to our captured scalp by the iterative closest point (ICP) method and project them onto the scalp. Starting from time t, we find velocities of all particles in the hair using the velocity field of time t, and then simulate the fiber motion by using the Back-Euler formula 3 or choosing a semi-Lagrangian scheme [SLF08] due to its efficiency and fast convergence. Once obtaining positions at $t+1$, we can access the according velocities in the velocity field of time $t+1$ to update or interpolate velocities in next step simulation. We show the capability of simulating complex hair styles using our reconstructed velocity field in the result section.

6. Results

In this section, we present the results of reconstructed hair dynamics from videos and the new hair animation driven by the captured velocity field. The whole sequence videos are contained in the supplemental material. All the system performance (in Table 1 2) are evaluated on a single quad-core PC of 3.20GHz CPU and 16GB memory.

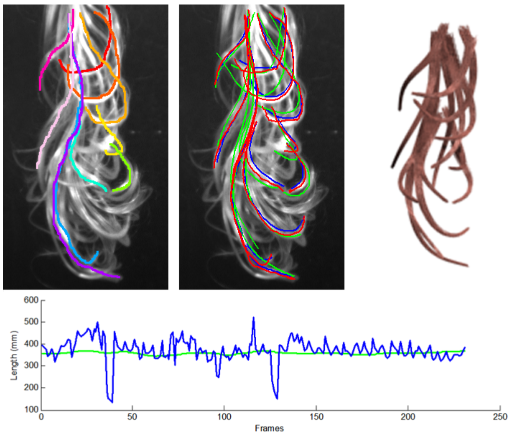
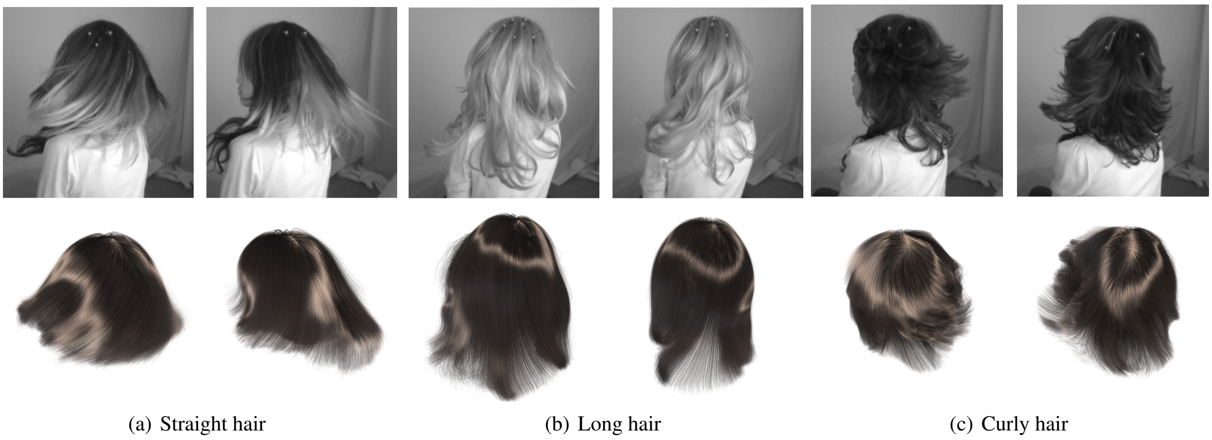


Figure 6: *The quantitative evaluation using manually labeled sktetches (top-left). The projections of the groundtruth, reconstructed and optimized fibers onto the image view are plotted in Red, Blue and Green respectively in the top-middle. The rendered 3D strands are shown in the top-right. The length change of the longest fiber is shown in the bottom before ($[132.8, 522.1]$mm) and after ($[345.3, 368.3]$mm) the optimization.*

Quantitative Evaluation We validate our local optimization algorithm using a small set of strands as shown in Figure 6. We manually sketch 11 strands of a 230-frame video sequence in three calibrated views and triangulate 3D fibers as the groundtruth fibers. In each view, 2D orientations are estimated near every sketch strand (less than 4 pixels) and the reconstructed 3D fibers grow from each sketch root point and terminate when 2D orientations are unavailable. We take reconstructed 3D fibers as inputs and compare the temporal optimized fibers with the groundtruth. The average distances of the reconstructed and optimized fibers to the groundtruth are 2.1021mm and 5.1722mm respectively. The length change of the longest fiber in the sequence is plotted in the bottom. Our optimization stabilizes the length oscillation of the image reconstructed fibers and also maintains close to the original shape. We randomly add offsets (less than 5mm) and render the 3D fibers in the right of Figure 6.

Full Hair Style We present three types of hair styles: the straight hair, the long straight hair with a wavy end, and the curly hair as shown in Figure 5. Each example approximately contains 7K to 8K hair fibers. The hair volume is built in a 220 × 190 × 210 regular grid and each grid cell is 2mm-by-2mm-by-2mm. The only user interaction is the manual correction of hair image mask and head pose tracking in the reconstruction stage. Even though they can be automatic generated by optical flow after initialized in the first


(a) Straight hair

(b) Long hair

(c) Curly hair

Figure 5: *The full swinging motion of three kinds of hair.*

frame, we still manually correct them every 30 frames to prevent drift.

The performance of each style is shown in Table 1. The initial image-based reconstruction step typically takes 15 to 25 minutes per frame. We take four processes in parallel and the total computation time of an entire sequence costs $15-16$ hours. In a local optimization iteration, each fiber takes an average of 6 seconds to optimize over a 200-frame sequence. The optimization of independent fibers are performed in parallel, we use 16 threads at a time and it takes approximate 2 hours to complete 7K fibers. The velocity diffusion step takes two minutes per frame on average for each style. Since we impose the local optimization to each individual fiber with limited storage, this step may be accelerated using graphic hardware in future work. We don't take into account of the fiber-body and fiber-fiber collision. According to our experiment, the motion of a long hair near the scalp surface follows the captured motion well and artifacts near hair tips are usually caused by the body collision and the missing of the reconstruction data. As we point out in our limitation, how to automatically balance appearance-smoothing and detail-preserving is still challenging.

New Hair Style We show two types of new hair styles in Figure 5. One is a "textured" curly hair style combined with the velocity field of the curly hair; The other one is a "helix" curly hair combined with the velocity of the long straight hair style. The performance is shown in Table 2.

To enhance the stability of hair simulation, we up-sampled the time step to 2×10^{-3}s, that is five times higher than our capture rate of velocity field. In the end, we only output in every five frames. Simulation combined with captured velocity field converges fast. Compared with the pure physics-based simulation, our method avoids the tedious manual adjustment of parameters to achieve the similar motion. The same artifact comes from the fiber-body collision since we don't model the neck and shoulder part, which will be allowed in our future performance capture work.

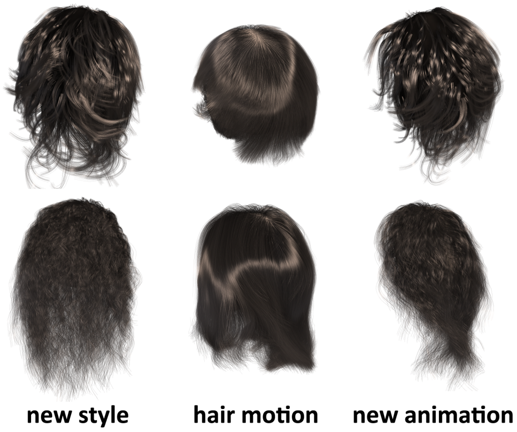


Figure 7: *The result of generating new hair animation using different hair styles. The static artist-made models are shown in the left column. The second row shows the reconstructed hair from videos. The right column shows the result of combining the artist-made hair with the hair velocity of the middle column.*

Limitation Image-based reconstruction of detailed hair styles is still a challenging problem, and our system is limited to the fact that it only refines dynamics and does not synthesize hair details. Ideally when sufficient hair details are provided from initial reconstruction results, our refinement algorithm is able to generate results with both details and physics accuracy. In practice, however, the quality of image-based static reconstruction of complicated hairstyles has room to improve.

7. Conclusion and Future Work

In this paper we present a hybrid framework to reconstruct *dynamic* hair models from multi-view videos. Our main contribution is a new refinement approach that combines physically based simulation with initially reconstructed hair models from images. Compared with pure physically based simulation approaches that usually require tedious and inherent manual parameters tuning to achieve a desired motion, our dynamic result can efficiently and naturally animate different full hair styles with fine details.

Looking into the future, we are interested in improving the refinement approach even further, so that it can handle more complicated cases with different hair styles. The initial image-based reconstruction greatly affects our final result quality, the numerical evaluation of image-based hair approach is still open and challenging, and we are looking for better solutions to this as well. Recovering from realistic rendered images of a synthetic data is also our future task to evaluate the approach. How to reduce the computational cost (by using graphics hardware for example) is another problem we would like to implement next.

Acknowledgements

This work is supported in part by University of Kentucky Research Foundation, US National Science Foundation award IIS-0448185, CPA-0811647, and Zhejiang Provincial Natural Science Foundation of China grant 2011C11053.

References

[BA93] BLACK M. J., ANANDAN P.: A framework for the robust estimation of optical flow. In *ICCV* (1993), pp. 231–236. 3

[BAC*06] BERTAILS F., AUDOLY B., CANI M.-P., QUERLEUX B., LEROY F., LÉVÊQUE J. L.: Super-helices for predicting the dynamics of natural hair. *ACM Trans. Graph. 25*, 3 (2006). 1, 2

[BAQ*05] BERTAILS F., AUDOLY B., QUERLEUX B., LEROY F., LÉVÊQUE J. L., AND M.-P. C.: Predicting natural hair shapes by solving the statics of flexible rods. *Eurographics (short papers)* (2005). 2

[Ber09] BERTAILS F.: Linear time super-helices. *Comput. Graph. Forum 28*, 2 (2009), 417–426. 2

[BW98] BARAFF D., WITKIN A.: Large steps in cloth simulation. In *SIGGRAPH* (1998), pp. 43–54. 4

[CCK05] CHOE B., CHOI M. G., KO H.-S.: Simulating complex hair with robust collision handling. In *SCA '05: Proceedings of the 2005 ACM SIGGRAPH/Eurographics symposium on Computer animation* (2005), pp. 153–160. 2

[DBDB11] DAVIET G., BERTAILS-DESCOUBES F., BOISSIEUX L.: A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. *ACM Trans. Graph. 30*, 6 (2011). 2

[Had06] HADAP S.: Oriented strands: dynamics of stiff multi-body system. In *ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2006), pp. 91–100. 2

[HMT01] HADAP S., MAGNENAT-THALMANN N.: Modeling dynamic hair as a continuum. *Comput. Graph. Forum 20*, 3 (2001). 2

[iAUK92] ICHI ANJYO K., USAMI Y., KURIHARA T.: A simple method for extracting the natural beauty of hair. In *SIGGRAPH* (1992), pp. 111–120. 2

[LLP*12] LUO L., LI H., PARIS S., WEISE T., PAULY M., RUSINKIEWICZ S.: Multi-view hair capture using orientation fields. In *Computer Vision and Pattern Recognition* (2012). 2

[LLW*11] LUO L., LI H., WEISE T., PARIS S., PAULY M., RUSINKIEWICZ S.: *Dynamic Hair Capture*. Tech. Rep. Technical Report TR-907-11, Princeton University, August 2011. 2

[MBTF03] MOLINO N., BRIDSON R., TERAN J., FEDKIW R.: A crystalline, red green strategy for meshing highly deformable objects with tetrahedra. In *12th Int. Meshing Roundtable* (2003), pp. 103–114. 2

[MSW*09] MCADAMS A., SELLE A., WARD K., SIFAKIS E., TERAN J.: Detail preserving continuum simulation of straight hair. *ACM Trans. Graph. 28* (July 2009), 62:1–62:6. 1

[MTPS04] MCNAMARA A., TREUILLE A., POPOVIĆ Z., STAM J.: Fluid control using the adjoint method. *ACM Trans. Graph. 23*, 3 (2004), 449–456. 5

[Pai02] PAI D. K.: Strands: Interactive simulation of thin solids using cosserat models. *Comput. Graph. Forum 21*, 3 (2002). 2

[PCK*08] PARIS S., CHANG W., KOZHUSHNYAN O. I., JAROSZ W., MATUSIK W., ZWICKER M., DURAND F.: Hair photobooth: geometric and photometric acquisition of real hairstyles. *ACM Trans. Graph. 27*, 3 (2008). 1, 2, 3, 4

[RCT91] ROSENBLUM R. E., CARLSON W. E., TRIPP E.: Simulating the structure and dynamics of human hair: Modelling, rendering and animation. *The Journal of Visualization and Computer Animation 2* (1991), 141–148. 2

[SLF08] SELLE A., LENTINE M., FEDKIW R.: A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3 (2008). 1, 2, 4, 6

[SSIF07] SIFAKIS E., SHINAR T., IRVING G., FEDKIW R.: Hybrid simulation of deformable solids. In *Eurographics* (2007), pp. 81–90. 2

[ST07] SPILLMANN J., TESCHNER M.: Corde: Cosserat rod elements for the dynamic simulation of one-dimensional elastic objects. In *Proc. ACM SIGGRAPH/Eurographics Symposium on Computer Animation* (2007), pp. 209–217. 2

[WBK*07] WARD K., BERTAILS F., KIM T.-Y., MARSCHNER S. R., CANI M.-P., LIN M.: A survey on hair modeling: Styling, simulation, and rendering. *IEEE Transactions on Visualization and Computer Graphics 13*, 2 (Mar-Apr 2007), 213–34. 2

[WLZ*09] WANG H., LIAO M., ZHANG Q., YANG R., TURK G.: Physically guided liquid surface modeling from videos. In *SIGGRAPH* (2009). 1

[WMT06] WOJTAN C., MUCHA P. J., TURK G.: Keyframe control of complex particle systems using the adjoint method. In *SCA* (2006). 5

[WOQyS05] WEI Y., OFEK E., QUAN L., YEUNG SHUM H.: Modeling hair from multiple views. In *SIGGRAPH* (2005), pp. 816–820. 1, 2, 3

[WYZG09] WANG L., YU Y., ZHOU K., GUO B.: Example-based hair geometry synthesis. *ACM Trans. Graph. 28*, 3 (2009). 4

[YWO09] YAMAGUCHI T., WILBURN B., OFEK E.: Video-based modeling of dynamic hair. In *PSIVT* (2009), pp. 585–596. 2, 3

[ZBLN94] ZHU C., BYRD R. H., LU P., NOCEDAL J.: *L-BFGS-B - Fortran Subroutines for Large-Scale Bound Constrained Optimization*. Tech. rep., ACM Trans. Math. Software, 1994. 5