Robust Hair Capture Using Simulated Examples

Liwen Hu*   Chongyang Ma*   Linjie Luo†   Hao Li*

*University of Southern California   † Adobe Research
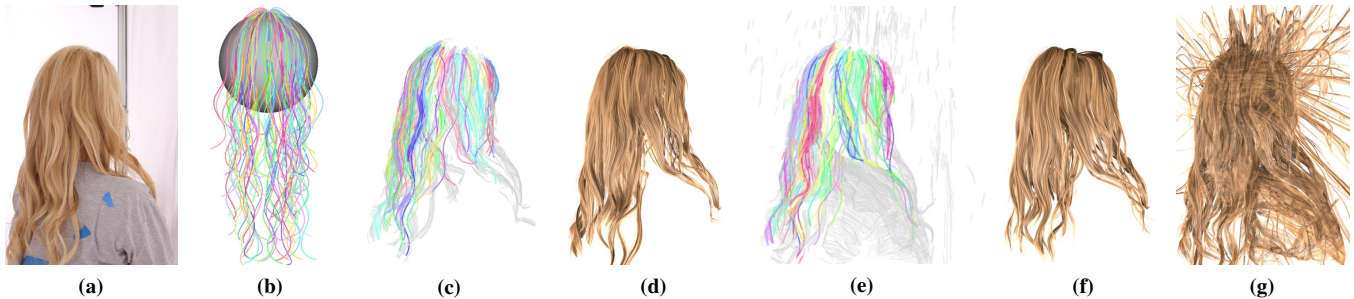
Figure 1: *Our system takes as input a few images (a) and employs a database of simulated example strands (b) to discover structurally plausible configurations from the reconstructed cover strands (c) for final strand synthesis (d). Our method robustly fits example strands to the cover strands which are computed from unprocessed outlier affected input data (e) to generate compelling reconstruction results (f). In contrast, the state-of-the-art method of [Luo et al. 2013a] fails in the presence of strong outliers (g).*

Abstract

We introduce a data-driven hair capture framework based on example strands generated through hair simulation. Our method can robustly reconstruct faithful 3D hair models from unprocessed input point clouds with large amounts of outliers. Current state-of-the-art techniques use geometrically-inspired heuristics to derive global hair strand structures, which can yield implausible hair strands for hairstyles involving large occlusions, multiple layers, or wisps of varying lengths. We address this problem using a voting-based fitting algorithm to discover structurally plausible configurations among the locally grown hair segments from a database of simulated examples. To generate these examples, we exhaustively sample the simulation configurations within the feasible parameter space constrained by the current input hairstyle. The number of necessary simulations can be further reduced by leveraging symmetry and constrained initial conditions. The final hairstyle can then be structurally represented by a limited number of examples. To handle constrained hairstyles such as a ponytail of which realistic simulations are more difficult, we allow the user to sketch a few strokes to generate strand examples through an intuitive interface. Our approach focuses on robustness and generality. Since our method is structurally plausible by construction, we ensure an improved control during hair digitization and avoid implausible hair synthesis for a wide range of hairstyles.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Geometric algorithms;

Keywords: hair capture, 3D reconstruction, data-driven modeling

Links: ◆DL 🅿PDF

1 Introduction

Just like in real life, hairstyles are essential elements for any digital character, reflecting personality, fashion, as well as one's cultural and social background. The generation of compelling 3D hair models in film or game production usually takes several weeks of manual work by a digital artist, involving the use of sophisticated hair modeling tools and procedural scripting. The task becomes even more tedious when capturing digital doubles, since the hair has to accurately match real references from photographs.

Unlike the 3D acquisition of objects with smooth surfaces (e.g., faces and bodies) which can usually take advantage of generic shape priors to regularize the reconstruction, it is generally very difficult to find a generic shape prior for hair because of the wide range of possible hairstyles. Specialized hair capture techniques are required due to extremely fine, intricate, and self-occluding structures.

Recently, graphics researchers have explored a variety of input sources for high-fidelity hair digitization ranging from multi-view stereo [Beeler et al. 2012], thermal imaging [Lay Herrera et al. 2012], depth-of-field [Jakob et al. 2009] to structured light [Paris et al. 2008]. Most of these approaches focus on matching the visual appearance through direct strand growing in a diffused orientation field and work well for sufficiently short and simple hair. For highly convoluted hairstyles, Luo et al. [2013a] lately demonstrated the importance of incorporating structural priors during the reconstruction process. Despite the effectiveness and accuracy, it depends on a good initial point cloud from multi-view stereo and uses a bottom-up strategy to connect local ribbons into wisps through purely geometry-inspired heuristics. Consequently, implausible hair strands that travel across different close wisps can still appear (Figure 10). Moreover, the approach relies on a careful manual clean-up procedure, involving tedious per-image hair segmentation and outlier removal after the initial point cloud reconstruction (see Figure 3). On average, segmenting the hair takes around 5 minutes per photograph, assisted with GrabCut [Rother et al. 2004], resulting in hours of work for a single hair reconstruction. Furthermore, Patch-based Multi-View Stereo (PMVS) [Furukawa and Ponce 2010] still produces a large mount of outliers due to ambiguous feature
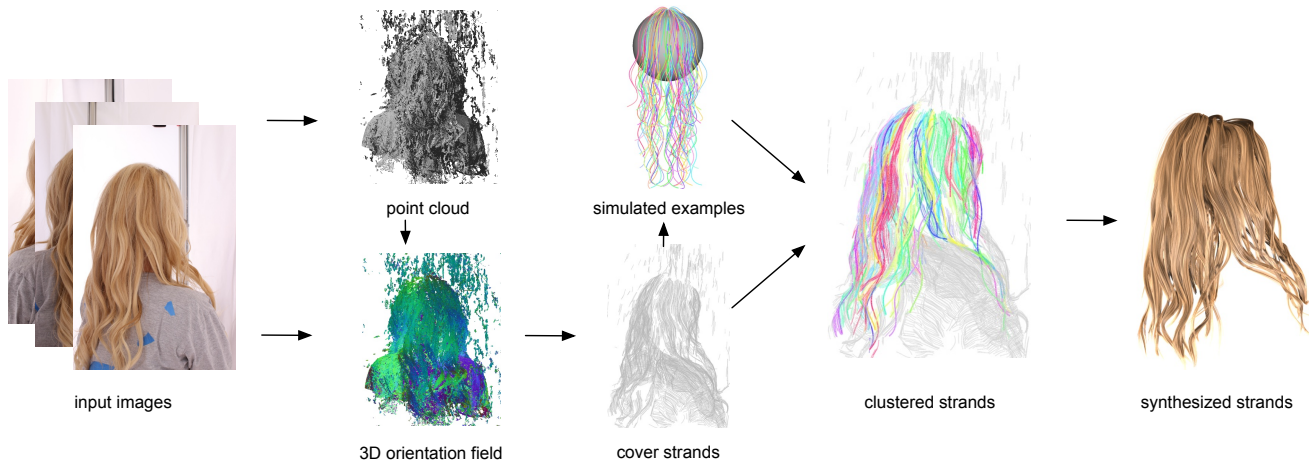
Figure 2: *The overview of our system. From a set of input images, we first reconstruct a point cloud as well as a 3D orientation field followed by a local growing step that generates the cover strands. We then generate a database of example strands through hair simulation constrained by the input hairstyle. We use this database to discover clustered strands from the cover with structurally plausible configurations via strand fitting and finally we synthesize realistic strands from these clustered strands.*

correspondences between the input images. A manual 3D outlier point removal procedure takes up to 30 minutes per hair example. A fully automatic end-to-end solution, from the capture session to the generation of a hair for proper simulation, has therefore not yet been proposed.

Figure 3: *Initial point cloud reconstructed from multi-view stereo with outliers.*

We develop a robust and general hair reconstruction framework that guarantees structural plausibility by construction using a database of strand examples obtained through simulation. While our method is designed for *unconstrained* hairstyles, we also allow user-sketched hair strands as database examples for some simple *constrained* styles (e.g., ponytail) which involve boundary constraints in addition to non-trivial inter-strand collisions and friction. During the reconstruction, we use our example strands in the database as *structural references* to ensure structural plausibility in the presence of large amount of outliers and occlusions in the input (Figure 1). To motivate, our database can constrain the range of possible strand lengths in the reconstruction, since this property is inherently difficult to estimate and enforce, especially for layered hairstyles with many crossing wisps.

For unconstrained hair, we generate structurally plausible strand examples through hair simulation based on the Super-Helices model introduced by Bertails et al. [2006]. To constrain the database within a manageable size, we reduce the simulation configuration space by a symmetry principle and constraints to the initial conditions, as well as a selection scheme for the natural parameters (e.g., stiffness and curliness) through the subsequent *strand fitting* step.

The strand fitting step discovers structurally plausible configurations of locally grown strand segments by a robust voting scheme to select matched examples while enforcing the roots of examples to lie close to the scalp. The best configurations are then extracted through mean-shift clustering. We then use these structural configurations to connect local strand segments into wisps in a top-down fashion for final hair strand synthesis.

Notice that the examples in our database do not need to match

the input exactly since we only use them as a structural references in subsequent steps. As a result, one key insight of our research is that *plausible structural configurations* can be encoded using a limited number of examples in a database (three orders of magnitude less than the total number of synthesized hair) to describe complex hairstyles that contain tens of thousands of strands with varying lengths and shapes.

While the fitting accuracy of our technique is on par with the state-of-the-art in 3D hair capture, our proposed method focuses on *robustness* and *generality*. Our system robustly generates structurally plausible hair from unprocessed point cloud input and handles a wide range of hairstyles. Due to the limited level of manual work required (if any, up to a few minutes for a novice user), we argue that our data-driven hair capture framework is suitable for efficient replication of complex hairstyles from the real world and can also be interesting for applications beyond digital production, such as virtual web content creation for cosmetic brands.

Contributions.

- A general top-down framework for robust 3D hair capture through a database of example strands. For unconstrained hair, only a limited set of examples based on a crude physical strand simulation is required. To handle constrained hair, our framework allows a few 2D user-sketched strokes.

- An efficient method to generate databases of example strands via physical simulation as priors of structural reference for hair reconstruction. We leverage symmetry and constrained initial conditions to significantly reduce the necessary number of simulation configurations.

- A robust fitting algorithm inspired by the Hough transform to discover structurally plausible hair configurations. Our method does not require any clean-up on the captured input data and can directly produce faithful hair models.

2 Related Work

Hair capture. Existing hair capture systems use various methods to acquire the orientation field and geometry for the exterior hair layer and rely on proper strand synthesis schemes to generate the final strands. Paris et al. [2004] introduced a method to estimate 3D

hair orientations from the hair's highlights under varying lighting and synthesize full strands from the scalp along these orientations. Wei et al. [2005] proposed a technique to compute 3D hair orientations from multi-view images by optimizing the orientation consistency across the views. Paris et al. [2008] developed an active acquisition system to accurately triangulate hair strand positions from projected light beams as well as a diffusion scheme to interpolate the orientation field for final strand synthesis. Jakob et al. [2009] demonstrated the possibility of using the shallow depth-of-field of a moving macro lens to localize in-focused hair strands from multiple views. Their system successfully captures strand-level geometry of loosely coupled hair assemblies. Chai et al. [2012] introduced a method to create approximate 3D strands from a single image that match the inter-strand occlusions and the fitted head model. This system has been recently extended to handle single video input to create hair animations [Chai et al. 2013], but is limited to relatively simple hairstyles. Beeler et al. [2012] introduced a multi-view stereo system to reconstruct short facial hair by strand-level matching. Their system uses a refinement step to improve strand connections and remove outliers. Herrera et al. [2012] pioneered a hair capture system using thermal imaging device. While impressive results were shown, their method is not suitable for hair strands that grow far away from the head. Luo et al. [2012; 2013b] developed several methods to reconstruct approximated surface-to-hair volume based on multi-view orientation fields. Recently, Luo et al. [2013a] proposed a structure-aware hair capture method that incorporates highly effective structural priors during the reconstruction process. The proposed system successfully handles highly convoluted wispy hairstyle as well as other non-trivial ones. Concurrent to our work, Echevarria et al. [2014] presented a method for physical reproduction of personalized figurines. Instead of reconstructing highly-detailed individual hair strands, they look for a printable 3D surface with stylized color and geometric details.

Hair simulation. A large category of hair simulation methods generates the motion of hair as an aggregated medium to reduce computational complexity such as with fluid continuum [Hadap and Magnenat-Thalmann 2001]. Techniques more relevant to our strand database generation are strand-level simulations based on strand dynamics. Many mechanical strand models have been investigated in the past. Selle et al. [2008] proposed a simple mass-spring model for efficient and robust hair simulation. The discrete elastic rods by [Bergou et al. 2008] provides an efficient simulation framework based on explicit centerline representation with reduced coordinates. Super-Helices [Bertails et al. 2006] use a piecewise helical discretization for the Cosserat rod model and allow efficient simulation with very few elements. We use this model to generate our simulated examples in the database because it is efficient and provides compelling natural hair simulation [Bertails et al. 2005].

Hair modeling. A general survey on existing hair modeling techniques can be found in [Ward et al. 2006]. Here, we only enumerate the most relevant ones. One class of hair modeling methods use physical simulation to generate hair strands such as [Bertails et al. 2005]. To better control the final shape of the hair, researchers also investigated other physically-inspired methods, e.g. vector fields [Yu 2001; Hadap and Magnenat-Thalmann 2000] and statistical wisp model [Choe and Ko 2005]. Sketch-based methods provide intuitive and convenient tools for hair modeling. Fu et al. [2007] demonstrated a sketch-based interface to build a vector field that generates the final hair strands. Wither et al. [2007] proposed a method to estimate hair simulation parameters from user strokes. Another category of hair modeling technique provides direct editing tools on hair geometry. These geometry-based methods use various structured models to facilitate controllable hair modeling. Kim and

Neumann [2002] employed a generalized cylinder hierarchy for multi-resolution hair editing. Ward et al. [2003] introduced a level-of-detail hair representation to ease hair modeling on different scales. Yuksel et al. [2009] demonstrated *hair meshes* to model complex hairstyles through topological operations on the polygonal meshes that drive the hairstyle. Wang et al. [2009] proposed a method to synthesize new hairstyles from input examples inspired by texture synthesis methods.

Model-based and example-based reconstruction. By knowing the scanned object in advance, devised parametric and procedurally generated models can be used for superior fitting control and robustness. Impressive reconstruction results can be achieved on a variety of objects such as trees [Livny et al. 2010], architectures [Nan et al. 2010], man-made objects [Li et al. 2010] and foliage [Bradley et al. 2013]. Example-based approaches leverages a set of related examples to improve the reconstruction from input data, typically of inferior quality. Convincing results have been demonstrated on part-based models [Shen et al. 2012], in-door furnitures [Shao et al. 2012] and models with only single image reference [Xu et al. 2011].

3 Overview

Our full pipeline is shown in Figure 2. Similar to [Beeler et al. 2012; Luo et al. 2013a], we use a multi-view stereo capture system for the input data. From the input photographs, we compute a rough initial 3D point cloud of the hair via Patch-based Multi-View Stereo (PMVS) algorithm [Furukawa and Ponce 2010]. For each point we compute the local 3D hair orientation field by maximizing the projected orientation consistency across the 2D orientation maps of each image. We use the 3D orientation field to generate a set of *cover strands* that are grown bidirectionally from each point as in Luo et al. [2013a]. The initial cover strands are mostly short and disconnected due to occlusions and missing data (Figure 4).

The example strands in the database are then generated through physical simulation based on Super-Helices [Bertails et al. 2005]. We simulate static hair strands under gravity and different boundary constraints (Section 4). Note that we do not aim to exactly reproduce the input hairstyle through simulation but rather use the example strands as structural references to discover plausible structural configurations among the cover strands. We exhaustively sample all possible simulation configurations within a pre-defined feasible parameter space constrained by the current input hairstyle. The final number of necessary simulations can be significantly reduced by leveraging symmetry principles and constraining the initial conditions. An optimal set of examples is then selected through an iterative scheme based on their fitting errors to the cover strands.

Next, we present a strand fitting scheme inspired by the Hough transform to simultaneously cluster and connect the cover strands into structurally plausible wisps based on the simulated examples (Section 5), instead of using separate steps of clustering and wisps connection as in Luo et al. [2013a]. We formulate a transformation space and ask each cover strand to cast votes for their *matched example strands* whose roots are constrained to the scalp model. The scalp model itself is obtained by fitting a 3D head model to the input point cloud [Li et al. 2013]. The matched examples are revealed as clustered modes in the transformation space and we use mean-shift clustering to extract these modes in an iterative fashion (Figure 6). The matched examples are then used to effectively discover the structural connections of fragmented cover strands from which we construct wisps to synthesize the final strands (Section 6).
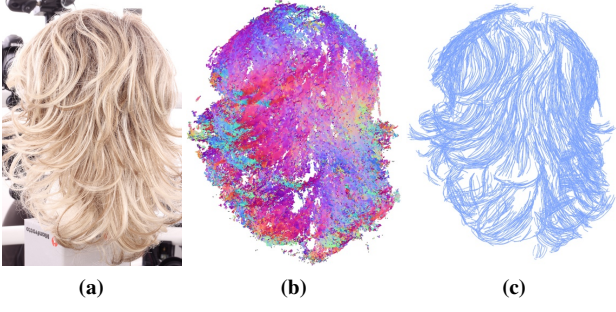
Figure 4: *Preprocessing steps. From a set of input images (a), our system first reconstructs a point cloud with a 3D orientation field (b). Cover strands are then grown to cover the entire point cloud based on the 3D orientation field (c).*
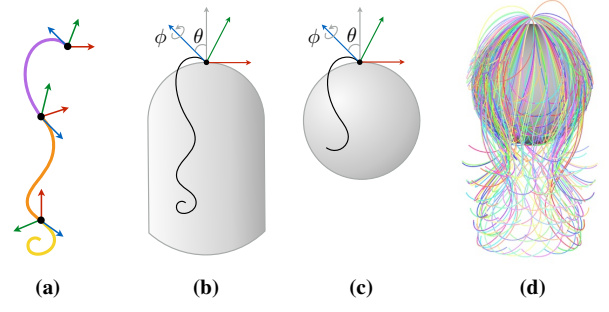

Figure 5: *The simulation setup. We use Super-Helices to simulate a natural hair strand which consists of a few piecewise helical segments (a). We use cylindrical (b) and spherical (c) boundary constraints for realistic simulation of strands with different lengths. A simulated database is shown in (d).*

4 Strand Simulation

Our goal in this section is to generate a database of strand examples $\{S_e\}$ for the subsequent strand fitting step (Section 5). We use the Super-Helices model [Bertails et al. 2005] to simulate static hair strands constrained by the scalp under gravity. With Super-Helices, a strand is modeled as a Cosserat rod, which is characterized by the following kinematics equation:

$$\frac{\partial \mathbf{e}_i}{\partial s} = \Omega \times \mathbf{e}_i, \qquad (1)$$

where $\{\mathbf{e}_i\}$ are the axes of the material frame $\mathbf{m}_1(s)$, $\mathbf{m}_2(s)$ and $\mathbf{t}(s)$ parametrized by curve length s for $i = 1, 2, 3$ respectively; $\Omega = \kappa_1(s)\mathbf{m}_1(s) + \kappa_2(s)\mathbf{m}_2(s) + \tau(s)\mathbf{t}(s)$ is the Daboux vector determined by the curvatures $\kappa_1(s)$, $\kappa_2(s)$ and torsion $\tau(s)$. Given the initial conditions, the curvatures and torsion uniquely determine the centerline of the rod $\mathbf{r}(s)$ through Equation 1 and $\mathbf{r}(s)$ can be integrated by the technique described in Bertails et al. [2006].

To simulate a static rod, one way is to minimize the potential energy. The potential energy $\mathcal{E}$ of a rod is given by:

$$\mathcal{E} = \mathcal{E}_e + \mathcal{E}_g + \mathcal{E}_c, \qquad (2)$$

where $\mathcal{E}_e$ is the internal elastic energy, $\mathcal{E}_g$ the gravitational potential energy, and $\mathcal{E}_c$ the collision energy.

To formulate $\mathcal{E}_e$, we first approximate the cross sections of the hair strands to be circular, despite the fact that the real hair strand's cross section varies among ethnic groups with up to 20% change in eccentricity. With this approximation, $\mathcal{E}_e$ can be written as:

$$\mathcal{E}_e = k \int_0^L \left(\kappa_1(s) - \kappa_1^0\right)^2 + \left(\kappa_2(s) - \kappa_2^0\right)^2 + A(\tau(s) - \tau^0)^2 ds,$$

where L is the strand length; κ_1^0, κ_2^0, and τ^0 are the natural curvatures and torsion of a hairstyle; k is the stiffness and we choose $A = 0.6$ since the Poisson ratio is constant across human hair.

We can directly rephrase $\mathcal{E}_g$ as follows:

$$\mathcal{E}_g = -B \int_0^L \mathbf{g} \cdot (\mathbf{r}(s) - \mathbf{r}(0)) ds$$
$$= -B \int_0^L (L - s)\mathbf{g} \cdot \mathbf{t}(s) ds$$

where $B = 4.1 \times 10^{-5}$ and $\mathbf{g}$ is the unit direction of gravity.

Finally, collision energy $\mathcal{E}_c$ can be modeled by elastic repelling force:

$$\mathcal{E}_c = D\Delta\ell^2$$

where D is set to 0.1 and $\Delta\ell$ is the amount of penetration of the rod from the colliding point on the scalp. Note that all the constants (A, B, D) above are derived from real physical quantities (see Appendix B in [Bertails et al. 2006]).

Therefore, given the boundary constraints, the simulation is determined by the initial conditions $\{\mathbf{m}_1(0), \mathbf{m}_2(0), \mathbf{t}(0), \mathbf{r}(0)\}$, strand length L, the natural curvatures and torsion $\kappa_1^0, \kappa_2^0, \tau^0$, and strand stiffness k. Since an exhaustive enumeration of the full simulation configuration space for a given hairstyle is intractable, one of our key insights is that we can leverage symmetry and constrained initial conditions to significantly reduce the number of simulation configurations. These simulated examples are used as rough structural references to ensure plausible hair structures on the cover strands for the final strand synthesis (Section 6).

To reduce the number of simulation configurations, we first simplify the boundary constraints. We approximate the scalp as a sphere so that simulations are equivalent on the same latitude as shown in Figure 5. However, for long hairstyles, other body parts (e.g., shoulders, neck, back) will affect the boundary constraints. Thus, we extrude the sphere model downwards to form a cylindrical shape to approximate these cases as shown in Figure 5b.

We further simplify our simulation setup by fixing $\mathbf{r}(0)$ on the top of the boundary model (Figure 5). In particular, simulating a strand from the top also covers the configurations that start from any middle point to the tip, using the material frame at the starting point as the initial condition. We also fix the angle θ between the normal of the top point and $\mathbf{t}(0)$ to 80 degrees since hair strands are not growing in parallel to the scalp (Figure 5).

With $\mathbf{r}(0)$ and $\mathbf{t}(0)$ fixed, we enumerate the last degree-of-freedom in the initial conditions by rotating the material frame around $\mathbf{t}(0)$ by $\phi \in [0, 2\pi)$ (Figure 5). To enumerate the strand length L, we uniformly sample L within the range of $100 \sim 400mm$ for all our datasets. Since the initial material frame rotates around $\mathbf{t}(0)$, only one of κ_1^0 and κ_2^0 is independent and we set $\kappa_2^0 = 0$. κ_1^0 is sampled within $0 \sim 0.1mm^{-1}$. The natural twist τ^0 occurs for african hair and other artificially permed hairstyles. For these hairstyles, we find it useful to set τ^0 within $-0.03 \sim 0.03mm^{-1}$. For most of the natural hairstyles, we set $\tau^0 = 0$ to enforce zero natural twisting according to [Bertails et al. 2006].

We generate the final hair databases by varying natural curvature κ_1^0 within the range of $[0.01, 0.06]$ with intervals of 0.01, and stiffness

k within the range of $[1.0, 2.0]$ with intervals of 0.5, based on the Young's modulus of natural hair [Bertails et al. 2006], resulting in a total number of $6 \times 3 = 18$ databases. Each database contains multiple strands by sampling ϕ and L on their feasible range. While in the worst case, we would iterate over each database to determine its fitness (Section 5), most databases can be excluded since they are significantly different than the target hairstyle (e.g., the bottom two hairstyles in Figure 15), resulting in much lower actual running times. In practice, we only need to select $1 \sim 5$ databases for each input hairstyle. Another possible way for parameter selection is to develop a scheme similar to [Wither et al. 2007] which estimates natural curvatures and stiffness from 2D user sketches.

5 Strand Fitting

Given a set of cover strands $\{S_c\}$ grown from the captured point cloud and orientation field (Figure 4), we first collect candidate rigid transformations $\{\mathbf{T}\}$ that are obtained by fitting those cover strands with example strands from the database $\{S_e\}$. In our case, a transformation has 7 components $\mathbf{T} = \{\mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z, \mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z, I\}$, where $(\mathbf{t}_x, \mathbf{t}_y, \mathbf{t}_z)^T$ is the translation, $\mathbf{r}_x, \mathbf{r}_y, \mathbf{r}_z$ the Euler angles, and I the index of the example strand in the database. The Euler angles are measured in radians and are multiplied by a factor of 10 to balance the weight of rotation vector with respect to translation vector. We incorporate the index values into the the transformation space for a unified treatment of extracting the *matched example strands*, which are the representatives of a cluster. We scale those index values by a factor of 100 as a sufficiently large value, so that the votes from different example strands will not affect each other and will be clustered separately.

We consider each pair of strands $\{S_c, S_e\}$ and compute the optimal transformation to align S_e with S_c. Specifically, we minimize the matching cost via Iterative Closest Point (ICP) algorithm with point-to-point constraints [Besl and McKay 1992]:

$$\mathcal{E}\big(S_c, \mathbf{T}(S_e)\big) = \sum_i \big|\mathbf{p}(s_{c,i}) - \mathbf{T}\big(\mathbf{p}(s_{e,i})\big)\big|^2 \qquad (3)$$

where $\{\mathbf{p}(s_{c,i})\}$ are the positions of all the samples in S_c, $s_{e,i}$ the closest sample of $s_{c,i}$ in S_e, and $\mathbf{T}\big(\mathbf{p}(s_{e,i})\big)$ the position of $s_{e,i}$ under the rigid translation and rotation specified by $\mathbf{T}$. We consider a transformation $\mathbf{T}$ as a valid vote to be collected, if and only if the matching cost defined by Equation 3 satisfies the following criterion:

$$\mathcal{E}\big(S_c, \mathbf{T}(S_e)\big) \leq T_{\mathcal{E}} \cdot n(S_c) \qquad (4)$$

where $T_{\mathcal{E}}$ is a user specified threshold and $n(S_c)$ the number of samples in the strand S_c. The threshold $T_{\mathcal{E}}$ controls the tolerance of average matching error between two strands, which essentially determines how many strands can be grouped into a single cluster. We fix this parameter to be $10mm$ for all of our datasets.

We estimate the hair root position using a RANSAC approach [Fischler and Bolles 1981]. More concretely, we first find the nearest point on the scalp for each cover strand, and randomly pick 5 nearby points within a radius of $20mm$ on the scalp as the potential roots. For each potential root, we attach it to the cover strand as an additional sample and run ICP for 30 iterations with each example strand to collect candidate transformations.

The collected transformations lie in a 7-dimensional parameter space. Intuitively, under the same or similar transformation, parallel cover strands which are close to each other can be aligned to the *same* part of a matched example strand. On the other hand, consistently oriented strands, that can be connected/merged into a single strand, would be aligned to *different* parts of the same example strand. As a result, the votes between cover strands and a matched example
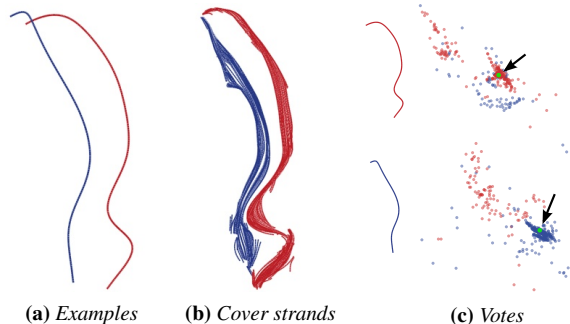

(a) *Examples*   **(b)** *Cover strands*   **(c)** *Votes*

Figure 6: *Illustration of strand fitting. With a database of two examples (a), two groups of cover strands (b) exhibit different behaviors in the votes (c), depending on the fitness of each example to those strands: the votes for fit examples are concentrated in clusters while the votes for unmatched examples are scattered and decentralized. The votes are projected onto the plane of their first two principal components for each example. The green dots (pointed out by the black arrows) show the cluster centers found by mean-shift.*

form a cluster in parameter space, while the votes between cover strands and non-matching examples are scattered (see Figure 6). We then perform mean-shift clustering to extract the most dominant clusters of candidate transformations in parameter space. While a similar approach has been proposed for symmetry detection of 3D shapes [Mitra et al. 2006], we instead group parallel strands together and simultaneously connect consistent strands. We propose to compute the local density ρ of each transformation $\mathbf{T}$ as a *scaled* sum of kernel functions K centered at $\mathbf{T}$:

$$\rho(\mathbf{T}) = C(\mathbf{T}) \cdot \sum_i K\big(|\mathbf{T} - \mathbf{T}_i| / \sigma\big) \qquad (5)$$

where σ is the kernel radius measured in the parameter space (we set $\sigma = 5$ for all the datasets used in the paper), and the scaling factor $C(\mathbf{T})$ is computed as the number of samples on the example strand that have been matched to certain samples on the cover strands. We choose the following kernel function as suggested by Comaniciu et al. [2002]:

$$K(x) = \begin{cases} 1 - x^2, & \text{if } 0 \leq x < 1 \\ 0, & \text{otherwise} \end{cases} \qquad (6)$$

The mean-shift clustering process is performed iteratively. Each time we extract the most dominant cluster mode which maximizes the kernel function in Equation 5 with the corresponding transformation $\mathbf{T}$. We apply the rigid translation and rotation to the example strand using $\mathbf{T}$, and reveal all the cover strands $\{S_c\}$ that satisfy the matching criterion defined by Equation 4. All those cover strands belong to the same cluster and the structure information is provided by the transformed example strand. We then remove all the votes which originate from those cover strands, and continue the clustering process until all the votes have been removed or enough number of clusters have been found.

6 Strand Synthesis

We synthesize final strands by independently considering each cluster extracted in the strand fitting stage as described in Section 5. Within each cluster, the corresponding cover strands $\{S_c\}$ serve as captured local features of the underlying hairstyle, while the matched example strand S_e provides the structural reference.

Similar to the synthesis algorithm described in [Luo et al. 2013a], we first group the cover strands $\{S_c\}$ within the same cluster into
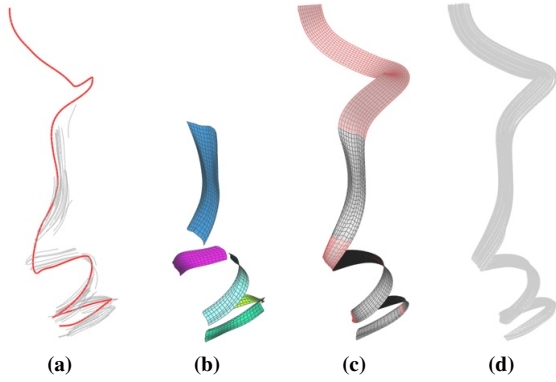
(a)   **(b)**   **(c)**   **(d)**

Figure 7: *Ribbon connection and strand synthesis. (a) Matched example strand is highlighted in red, cover strands in the same cluster are highlighted in blue. (b) Cover strands are grouped into multiple ribbons. (c) We connect two ribbons based on the correspondence between the example strand and the ribbons. Here the segment of samples in the example strand which are not matched to any samples of the ribbon centers are highlighted in red. (d) Final output strands.*

multiple ribbons, where each ribbon is a flat surface fragment composed of close and nearly parallel strands. Then we connect the ribbons to form wisps and synthesize output strands from those wisps. The major difference between Luo et al. [2013a] and our strand synthesis approach is that, instead of using some heuristics like local circle fitting which cannot ensure global structural plausibility, we use the structure information provided by the example strand to connect those ribbons.

Ribbon Connection. We first build the correspondence between the example strand and the center strands of all the ribbons within a single cluster. For each sample on a center strand, we first find the closest sample on the example strand. A ribbon is considered to be matched to the segment $[i_{\min}, i_{\max}]$ of the example strand, when $i_{\min}$ and $i_{\max}$ are the minimum and maximum indices of the closest samples. We then divide those ribbons into several subsets while ensuring that (1) each ribbon belongs to at least one subset; (2) the matched segments within the same subset do not overlap; (3) the total length of the matched segments within each segment is maximized by including as many ribbons as possible. For the ribbons in each subset, we resample them by linearly interpolating original sample positions, so that they have the same ribbon width w as the maximum width before resampling, and the ribbon length l equals to $(i_{\max} - i_{\min} + 1)$. Finally we sort those ribbons in the same subset based on the indices of matched segments, and connect them one by one to get a wisp of width w. The wisp length l is the same as the example strand (see Figure 7).

For the samples $\{s\}$ in the example strand which are matched to certain samples on the ribbon centers, we use the resampled ribbon positions for the wisp. For those segments of successive samples $\{s_i\}$ in the example strand which are not matched to any samples of the ribbon centers, we determine their new positions $\{\mathbf{p}(s_i)\}$ in the wisp by minimizing the following energy for each segment of length N:

$$\mathcal{E}\big(\{\mathbf{p}(s_i)\}\big) = \sum_{i=1}^{N-1} \big|\big(\mathbf{p}(s_i) - \mathbf{p}(s_{i+1})\big) - \big(\tilde{\mathbf{p}}(s_i) - \tilde{\mathbf{p}}(s_{i+1})\big)\big|^2$$
$$+ \alpha \sum_{i=1}^{N} |\mathbf{p}(s_{i-1}) - 2\mathbf{p}(s_i) + \mathbf{p}(s_{i+1})|^2 \qquad (7)$$

Figure 8: *Our capture setup.*

where $\tilde{\mathbf{p}}(s_i)$ is the original position of sample s_i in the example strand and α a user specified weight (we use $\alpha = 5$ for all the results throughout the paper). For the second term in Equation 7, we extend the segment by predefining $\mathbf{p}(s_0)$ and $\mathbf{p}(s_{N+1})$ to be the two positions of the ribbon samples adjacent to this segment, if either one of the two samples s_0 and s_{N+1} exists. This term ensures that the unmatched segments of the example strand can be smoothly deformed into a new segment for ribbon connections. The final output strands are synthesized from the wisps using the method of Luo et al.[2013a].

7 Results

We test the performance of our hair capture framework on different data sets which have different multi-view stereo configurations:

- Set A includes the Figures 1 and 9 and was captured using 66 synchronized Canon EOS Rebel T3i cameras (864×1296 pixels, focal length of 30 mm). The capture setting is shown in Figure 8 and all subjects were real people.

- Set B is shown in Figure 15 (top three rows) and used 45 Canon EOS 550D cameras (864×1296 pixels, focal length of 50 mm) to capture static wigs.

- Set C, illustrated in Figure 15 (bottom), contains the wig from the motorized gantry setting of [Luo et al. 2013a] where 50 shots around the hair were taken (936×1404 pixels and 50 mm lens).

As demonstrated in Figure 1f, our method effectively reconstructs complex and curly hairstyles (Set A) on fully unprocessed input point clouds with many outliers and the presence of surface geometries from the face and body, while the method of Luo et al. [2013a] breaks, as shown in Figure 1g.

The first two rows of Figure 15, show examples of wigs that contain many crossing wisps and hair strands of varying lengths. The multiple layers of occluding hair also increase the difficulty of estimating connections. We show that using a database with simulated strands is effective in discovering the connections of fragmented hair wisps and producing structurally plausible hair configurations during hair synthesis. The third row in Figure 15 shows an example with straight hair where a database with minimal number of example strands is sufficient to encode global properties of the hair structure.

As shown on the challenging example of Set C (Figure 15, bottom), our method produces results that are visually as compelling as the current state-of-the-art with faithfully reconstructed ringlet structures.
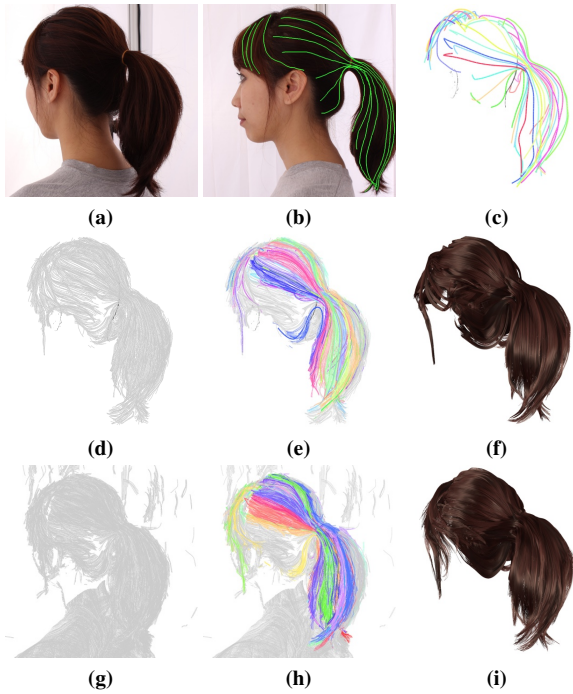
Figure 9: *Reconstruction results using examples from user strokes. From an input photo (a) of a constrained hairstyle, the user can draw strokes (b) for a set of plausible strands. Our system creates 3D strand examples (c) from these strokes to fit the input cover strands as shown in (d) and (g) without and with outliers respectively. The strand fitting step discovers clustered strands (e) / (h) which we use to synthesize final strands (f) / (i).*

Reference photo   [Luo et al. 2013a]   Our result

Figure 10: *Comparison between [Luo et al. 2013a] and our reconstruction result in terms of structural plausibility.*

To further demonstrate the generality of our solution, we show in Figure 9 that simple constrained hairstyles (e.g., ponytail) can be digitized when we replace the simulated database with examples from user sketches. The user only needs to draw a few strokes on the input photo to depict plausible structure of the underlying hairstyle
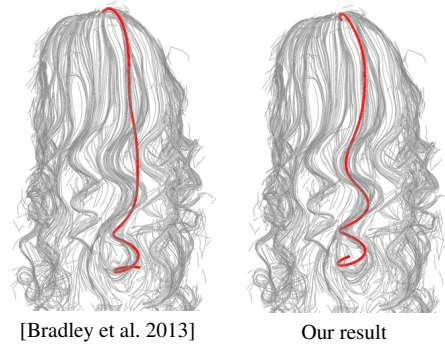

[Bradley et al. 2013]   Our result

Figure 11: *Comparison between direct fitting based on RANSAC [Bradley et al. 2013] and our voting-based fitting method. The direct fitting method is often stuck in local minima and finds sub-optimal matched example.*

(see Figure 9b), which can be done in minutes. We then re-project those 2D strokes back to the surface of the captured point cloud and smooth the strands using Equation 7 with an additional positional term to get a database of 3D examples (see Figure 9c). Notice that strands from user sketches, simulation, or other sources can be combined to form a more powerful prior, but were not required in any of our examples.

Comparison. We compare our method with the state-of-the-art hair capture technique of [Luo et al. 2013a]. As illustrated in Figure 10, our method is often better at discovering correct inter-wisp connections than the bottom-up approach of Luo et al. [2013a].

We also compare our fitting algorithm based on mean-shift clustering (Section 5) with a direct fitting approach based on RANSAC, as described in Bradley et al. [2013]. While their method is designed for foliage reconstruction, we adopt its matching strategy to hair data, as shown in Figure 11. We initialize each example strand with random rigid transformations and perform ICP to find the best alignment to the point cloud. We found that the aligned example strands using the smallest ICP fitting error are often local minima (Figure 11, left) as opposed to our voting approach (Figure 11, right).

Evaluation. We evaluate the robustness of our reconstruction algorithm using databases of decreasing sizes as well as of different simulation parameters. In Figure 12, we reduce the samples of ϕ within $[0, 2\pi)$ by factors of two, given each strand length L, resulting in databases with 256, 128 and 64 examples. In Figure 13, we use two different databases with $k = 2.0$ and $k = 1.5$ respectively to reconstruct the same hairstyle. We also evaluate the robustness of our algorithm against outliers in the input as shown in Figures 1 and 9. Our method can faithfully recover almost all hair structures with little visual difference to the cleaned-up results without outliers.

Simulation. To simulate our resulting hairstyles, individual fibers were first converted into piecewise helices (with 8 helical arcs each) using the floating tangents algorithm [Derouet-Jourdan et al. 2013] that served to initialize the rest shape of an assembly of super-helices [Bertails et al. 2006]. The resulting physical hair model was then animated under wind forces using Daviet et al.'s simulation solver [Daviet et al. 2011] for computing hair-hair and hair-body frictional contacts.

Parameters and timings. While the capture is instantaneous (one shot), the entire pipeline runs at about $1 \sim 2$ hours for each dataset on a modern machine (2.6 GHz Intel Core i7 with 16 GB RAM).
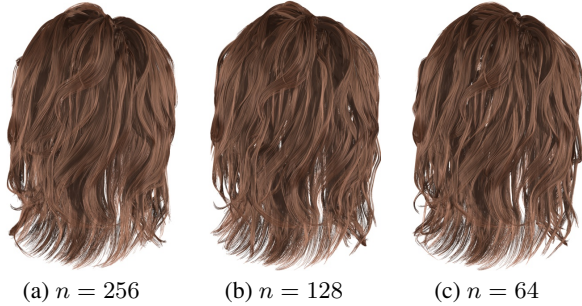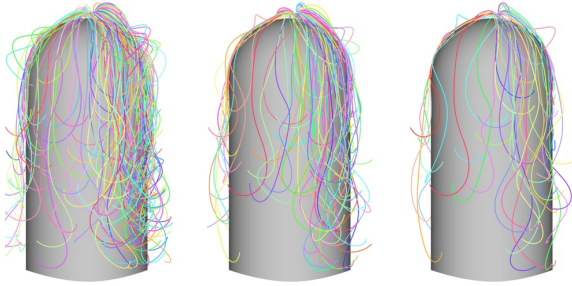
Dataset	Example strand #	κ_1^0	τ^0	k	Cover strand #	Output strand #
Fig. 1d	112	0.03	± 0.03	1.0	2973	$49k$
Fig. 1f	112	0.03	± 0.03	1.0	5860	$47k$
Fig. 15, 1st	256	0.05	0	1.0	4473	$38k$
Fig. 15, 2nd	208	0.04	0	2.0	3801	$56k$
Fig. 15, 3rd	20	0.02	0	1.5	700	$46k$
Fig. 15, 4th	256	0.06	0	2.0	2624	$45k$
Fig. 9	38	N/A	N/A	N/A	2356	$74k$
Fig. 12b	128	0.05	0	1.0	5860	$32k$
Fig. 12c	64	0.05	0	1.0	5860	$34k$

Table 1: *Parameter settings and statistics for our results.*

(a) $n = 256$ (b) $n = 128$ (c) $n = 64$

Figure 12: *Comparison of reconstruction results using different numbers of example strands.*
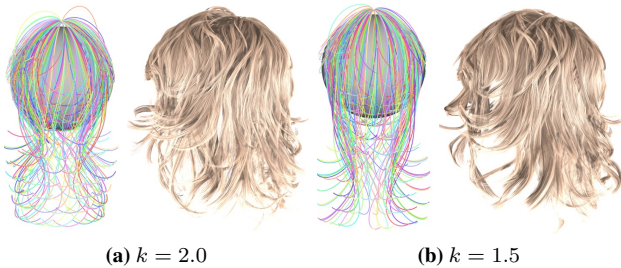

(a) $k = 2.0$ (b) $k = 1.5$

Figure 13: *Comparison of reconstruction results using databases with different simulation parameters.*

Figure 14: *Results of dynamic simulation using our output strands.*

The physical simulation takes about 5 minutes for a database of 256 examples. The computation of the hair fitting stage takes $10 \sim 40$ minutes, depending on the size of the database and cover strands. See Table 1 for parameters and statistics of all the datasets.

8 Conclusion

We have demonstrated that a data-driven fitting strategy with physics-driven hair strands can effectively discover plausible structural configurations of unconstrained but highly complex hairstyles. We found that only a crude physics simulation is needed to reliably bridge the connection between fragmented cover strands. Moreover, very small number of example strands are necessary for the reconstructions and sufficient to handle a wide range of complex hairstyles. Our ponytail example shows that simple constrained hairstyles can be treated by allowing rough user sketches instead of a simulation. While we are still far from a fully accurate strand-level reconstruction and faithful estimate of hair lengths, the results of our approach is perceptually on par with the state-of-the-art. Because we impose a stronger global prior on structure, the local wisps may be more accurately captured using the approach of [Luo et al. 2013a]. Nevertheless, our method facilitates accessibility and deployment as it is resistant to imperfect and unprocessed input data. In particular, our approach is capable of producing a wider range of hairstyles with a more effective control over global structural plausibility than previous methods.

Limitations. As shown in Section 7, our example-based reconstruction method can robustly handle a wide range of complex hairstyles. However, since our simulation method assumes that the captured hairstyle has uniform natural stiffness and curliness, we cannot generate realistic simulation for hairstyles that involve artificial styling processes such as perming and gluing. One possible future research direction is to extend our current simulation model to cover hairstyles with varying physical parameters. This also leads to an investigation on efficient methods to automatically infer realistic simulation parameters from incomplete cover strands.

Currently, the strand fitting step is the main bottleneck of our framework. Despite the quality of the fitting results, the computations can take more than an hour even with multi-threading. We would like to investigate more effective shape analysis and matching algorithms to accelerate the strand fitting process.

Future Work. Since our solution involves minimal user input, it allows a massive collection of a wide variety of plausible hairstyles. Such hair collection can be useful for further explorations of a possible general database that can cover the space of all possible hairstyles, potentially improving the performance of data-driven hair capture techniques and impacting research for hairstyle learning and classification. We also plan to investigate the use of procedural priors to capture even more challenging constrained hairstyles such as braided ones, updos, and afros. Ultimately, we wish to replace our multi-view stereo capture system with cost-effective and low-impact sensing devices for deployment to consumer-grade applications.

Acknowledgements

Figure 15: *Our reconstruction results. For each dataset, we show reference photo, input point cloud with RGB color-encoded 3D orientation field, database of simulated examples, top 10 clustered strands, our final synthesized result and comparative result from [Luo et al. 2013a].*

References

BEELER, T., BICKEL, B., NORIS, G., MARSCHNER, S., BEARDS-LEY, P., SUMNER, R. W., AND GROSS, M. 2012. Coupled 3D reconstruction of sparse facial hair and skin. *ACM Trans. Graph. 31*, 117:1–117:10.

BERGOU, M., WARDETZKY, M., ROBINSON, S., AUDOLY, B., AND GRINSPUN, E. 2008. Discrete elastic rods. *ACM Trans. Graph. 27*, 3, 63:1–63:12.

BERTAILS, F., AUDOLY, B., QUERLEUX, B., LEROY, F., LEV-EQUE, J.-L., AND CANI, M.-P. 2005. Predicting natural hair shapes by solving the statics of flexible rods. In *Eurographics (short papers)*, 81–84.

BERTAILS, F., AUDOLY, B., CANI, M.-P., QUERLEUX, B., LEROY,

F., AND LÉVÊQUE, J.-L. 2006. Super-helices for predicting the dynamics of natural hair. *ACM Trans. Graph. 25*, 3, 1180–1187.

BESL, P., AND MCKAY, N. D. 1992. A method for registration of 3-d shapes. *IEEE Trans. on PAMI 14*, 2, 239–256.

BRADLEY, D., NOWROUZEZAHRAI, D., AND BEARDSLEY, P. 2013. Image-based reconstruction and synthesis of dense foliage. *ACM Trans. Graph. 32*, 4, 74:1–74:10.

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

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

CHOE, B., AND KO, H.-S. 2005. A statistical wisp model and pseudophysical approaches for interactive hairstyle generation. *TVCG 11*, 2, 160–170.

COMANICIU, D., AND MEER, P. 2002. Mean shift: a robust approach toward feature space analysis. *IEEE Trans. on PAMI 24*, 5, 603–619.

DAVIET, G., BERTAILS-DESCOUBES, F., AND BOISSIEUX, L. 2011. A hybrid iterative solver for robustly capturing coulomb friction in hair dynamics. *ACM Trans. Graph. 30*, 6, 139:1–139:12.

DEROUET-JOURDAN, A., BERTAILS-DESCOUBES, F., AND THOLLOT, J. 2013. Floating tangents for approximating spatial curves with G^1 piecewise helices. *Computer Aided Geometric Design 30*, 5, 490–520.

ECHEVARRIA, J. I., BRADLEY, D., GUTIERREZ, D., AND BEELER, T. 2014. Capturing and stylizing hair for 3D fabrication. *ACM Trans. Graph. 33*, 4.

FISCHLER, M. A., AND BOLLES, R. C. 1981. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM 24*, 6, 381–395.

FU, H., WEI, Y., TAI, C.-L., AND QUAN, L. 2007. Sketching hairstyles. In *SBIM '07*, 31–36.

FURUKAWA, Y., AND PONCE, J. 2010. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. PAMI 32*, 1362–1376.

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

HADAP, S., AND MAGNENAT-THALMANN, N. 2001. Modeling dynamic hair as a continuum. *Computer Graphics Forum 20*, 3, 329–338.

JAKOB, W., MOON, J. T., AND MARSCHNER, S. 2009. Capturing hair assemblies fiber by fiber. *ACM Trans. Graph. 28*, 5, 164:1–164:9.

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

LAY HERRERA, T., ZINKE, A., AND WEBER, A. 2012. Lighting hair from the inside: A thermal approach to hair reconstruction. *ACM Trans. Graph. 31*, 6, 146:1–146:9.

LI, G., LIU, L., ZHENG, H., AND MITRA, N. J. 2010. Analysis, reconstruction and manipulation using arterial snakes. *ACM Trans. Graph. 29*, 5, 152:1–152:10.

LI, H., YU, J., YE, Y., AND BREGLER, C. 2013. Realtime facial animation with on-the-fly correctives. *ACM Trans. Graph. 32*, 4, 42:1–42:10.

LIVNY, Y., YAN, F., OLSON, M., CHEN, B., ZHANG, H., AND EL-SANA, J. 2010. Automatic reconstruction of tree skeletal structures from point clouds. *ACM Trans. Graph. 29*, 6, 151:1–151:8.

LUO, L., LI, H., PARIS, S., WEISE, T., PAULY, M., AND RUSINKIEWICZ, S. 2012. Multi-view hair capture using orientation fields. In *CVPR*, 1490–1497.

LUO, L., LI, H., AND RUSINKIEWICZ, S. 2013. Structure-aware hair capture. *ACM Trans. Graph. 32*, 4, 76:1–76:12.

LUO, L., ZHANG, C., ZHANG, Z., AND RUSINKIEWICZ, S. 2013. Wide-baseline hair capture using strand-based refinement. In *CVPR*, 265–272.

MITRA, N. J., GUIBAS, L. J., AND PAULY, M. 2006. Partial and approximate symmetry detection for 3D geometry. *ACM Trans. Graph. 25*, 3, 560–568.

NAN, L., SHARF, A., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2010. SmartBoxes for interactive urban reconstruction. *ACM Trans. Graph. 29*, 4, 93:1–93:10.

PARIS, S., BRICEÑO, 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. *ACM Trans. Graph. 27*, 3, 30:1–30:9.

ROTHER, C., KOLMOGOROV, V., AND BLAKE, A. 2004. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans. Graph. 23*, 3, 309–314.

SELLE, A., LENTINE, M., AND FEDKIW, R. 2008. A mass spring model for hair simulation. *ACM Trans. Graph. 27*, 3, 64:1–64:11.

SHAO, T., XU, W., ZHOU, K., WANG, J., LI, D., AND GUO, B. 2012. An interactive approach to semantic modeling of indoor scenes with an rgbd camera. *ACM Trans. Graph. 31*, 6, 136:1–136:11.

SHEN, C.-H., FU, H., CHEN, K., AND HU, S.-M. 2012. Structure recovery by part assembly. *ACM Trans. Graph. 31*, 6, 180:1–180:11.

WANG, L., YU, Y., ZHOU, K., AND GUO, B. 2009. Example-based hair geometry synthesis. *ACM Trans. Graph. 28*, 3, 56:1–56:9.

WARD, K., LIN, M. C., LEE, J., FISHER, S., AND MACRI, D. 2003. Modeling hair using level-of-detail representations. In *Proc. CASA*, 41–47.

WARD, K., BERTAILS, F., YONG KIM, T., MARSCHNER, S. R., PAULE CANI, M., AND LIN, M. C. 2006. A survey on hair modeling: Styling, simulation, and rendering. *TVCG 13*, 2, 213–234.

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

WITHER, J., BERTAILS, F., AND CANI, M.-P. 2007. Realistic hair from a sketch. In *SMI '07*, 33–42.

XU, K., ZHENG, H., ZHANG, H., COHEN-OR, D., LIU, L., AND XIONG, Y. 2011. Photo-inspired model-driven 3D object modeling. *ACM Trans. Graph. 30*, 4, 80:1–80:10.

YU, Y. 2001. Modeling realistic virtual hairstyles. In *Pacific Graphics'01*, 295–304.

YUKSEL, C., SCHAEFER, S., AND KEYSER, J. 2009. Hair meshes. *ACM Trans. Graph. 28*, 5, 166:1–166:7.