

# Single-View Hair Modeling Using A Hairstyle Database

Liwen Hu\*

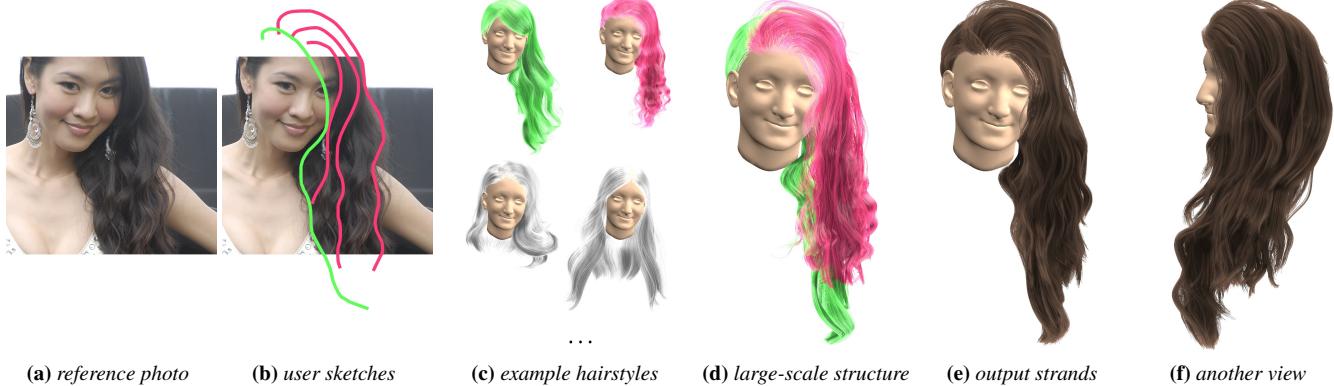
Chongyang Ma\*

Linjie Luo<sup>†</sup>

Hao Li\*

\*University of Southern California

<sup>†</sup>Adobe Research



**Figure 1:** Our system takes as input a reference photo (a), a few user strokes (b) and a database of example hairstyles (c) to model the 3D target hairstyle (e). The retrieved examples that best match the user strokes are highlighted with the corresponding colors in (c) which are combined consistently as shown in (d). Original image courtesy of Yung-Yuan Kao.

## Abstract

Human hair presents highly convoluted structures and spans an extraordinarily wide range of hairstyles, which is essential for the digitization of compelling virtual avatars but also one of the most challenging to create. Cutting-edge hair modeling techniques typically rely on expensive capture devices and significant manual labor. We introduce a novel data-driven framework that can digitize complete and highly complex 3D hairstyles from a single-view photograph. We first construct a large database of manually crafted hair models from several online repositories. Given a reference photo of the target hairstyle and a few user strokes as guidance, we automatically search for multiple best matching examples from the database and combine them consistently into a single hairstyle to form the large-scale structure of the hair model. We then synthesize the final hair strands by jointly optimizing for the projected 2D similarity to the reference photo, the physical plausibility of each strand, as well as the local orientation coherency between neighboring strands. We demonstrate the effectiveness and robustness of our method on a variety of hairstyles and challenging images, and compare our system with state-of-the-art hair modeling algorithms.

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

**Keywords:** hairstyle database, data-driven modeling, sketch-based retrieval, structure-aware shape processing, piecewise helices

## 1 Introduction

With the dramatic performance increase of today’s graphics technologies, visual details of digital humans in games, online virtual worlds, and virtual reality applications are becoming significantly more demanding, reaching nearly the quality and realism seen in film production. While compelling animations and renderings are now possible in realtime, the digitization of lifelike virtual avatars is still reserved to professional production studios, involving complex hardware equipment and talented digital artists. With the aim of scaling production and bringing 3D character creation to consumers, significant research has been dedicated to the automatic digitization of human faces [Blanz and Vetter 1999], bodies [Zhou et al. 2010], and hands [Wang and Popović 2009] from a single input image or video. These 3D reconstruction methods typically rely on low dimensional parametric models that encode statistical shape variations of a population.

An essential but often overlooked problem is the modeling of human hair, which is challenged by the highly intricate structure of intertwined strands and the wide variation of hairstyles. Unlike anatomically compatible shapes such as faces and bodies, it is unclear how to represent or construct a unified parametric model for hair due to the vast diversity in topological structures and local details. Nevertheless, several 3D hair capture techniques have recently demonstrated the successful reconstruction of challenging hairstyles [Paris et al. 2008; Luo et al. 2013], but they typically rely on complex capture settings. Although single-view hair modeling methods [Chai et al. 2012; Chai et al. 2013] have been introduced lately for image-based rendering and editing tasks, they are only capable of handling very coarse and smooth hairstyles.

Recent advances in data-driven modeling have shown that many ill-posed single-view reconstruction problems can be addressed by leveraging geometric priors from a large collection of 3D shapes (e.g., for hand tracking [Wang and Popović 2009], pose estimation

[Shotton et al. 2011], and scan completion [Shen et al. 2012]). The main challenge of data-driven methods generally consists of maximizing the coverage of all input cases while keeping the solution space of the database tractable since acquiring, storing, and searching the database becomes more expensive as the database size increases. A common strategy is to decompose each example into components which can be recombined into new examples at runtime. While Shen et al. [2012] have demonstrated the power of such methodology on the problem of scan completion of man-made objects that are composed of individual components (e.g., seat, arms and legs of chairs), it is generally unclear how to decompose a hairstyle into semantically meaningful and discernible parts since hair structures can be cut, split and grouped in almost arbitrary ways.

In this work, we present a system that creates a high quality 3D hair model from a single input reference photo using a database of full 3D hairstyles and a few user strokes. Our hairstyle database is obtained by collecting more than 300 hairstyles which are created manually and publicly available on some online game communities [Electronic Arts 2014; Newsea 2014]. We propose a coarse-to-fine modeling strategy and adopt a hierarchical two-level representation to accurately model the target hairstyle.

At the coarse scale, we first retrieve the example hairstyles from the database which best match the 2D user strokes. These strokes outline the representative structures in the target hairstyle which are necessary for revealing the full hair connectivity and topology, often occluded in single-view images. We then combine the retrieved examples into a 3D orientation field by considering both the user strokes and local orientation consistency in a Markov Random Field (MRF) framework. We use the combined 3D orientation field to produce the structure of the target hairstyle by growing strands from the scalp of a fitted 3D head model. At the fine scale, we first deform the combined hair strands to match the 2D orientations of the target hairstyle. To produce physically plausible hair strands during the refinement, one possibility is to simulate the strands. However, while inferring simulation parameters has been demonstrated on 3D data [Derouet-Jourdan et al. 2013], those variables are extremely difficult to estimate from 2D input. Since hair strands can be represented by piecewise helices [Bertails et al. 2006], we adopt the method proposed by Cherin et al. [2014] to fit piecewise 3D helix curves to the 2D projections of the target hair strands. Finally we refine the output strands by jointly optimizing the similarity to these fitted helices as well as the orientation consistency between neighboring strands.

We demonstrate the effectiveness and robustness of our approach through a variety of examples (Section 6). We also compare our method with state-of-the-art hair modeling techniques (Figure 8). In summary, our main contributions are:

- The first hair modeling system based on a database of complete hairstyles which is capable of creating high quality 3D hair models from a single image with minimal user input.
- A hair structure assembly algorithm that can combine multiple hairstyles into a consistent one to match a target hairstyle.
- A hair strand synthesis method to refine a coarse 3D hairstyle by matching the projected 2D local orientation of the target hairstyle with physically plausible strands.

## 2 Previous Work

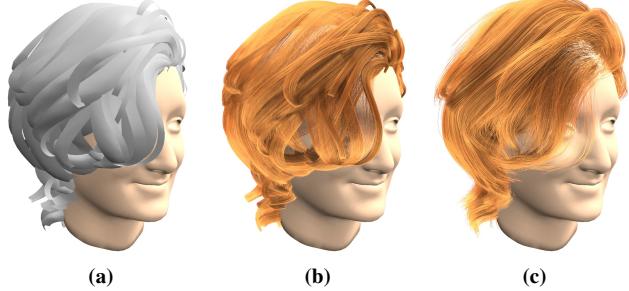
**Hair digitization.** The modeling of human hair has been extensively explored in graphics research offering a wide range of sophisticated and intuitive design tools [Kim and Neumann 2002; Choe and Ko 2005; Ward et al. 2007; Yuksel et al. 2009; Weng et al. 2013]. More recently, 3D hair capture techniques have

been introduced to digitize physical hair through optical sensing. In general, these methods are often associated with expensive hardware equipment, controlled capture settings, and tedious manual segmentation tasks [Paris et al. 2008; Jakob et al. 2009; Lay Herrera et al. 2012; Luo et al. 2013; Echevarria et al. 2014; Xu et al. 2014b]. To fully eliminate the manual segmentation process, Hu et al. [2014a] developed a hair capture technique based on simulated hair strands which enables the automatic segmentation of hair regions and the synthesis of physically plausible hair models. This method is only suitable for unconstrained hairstyles and also require a costly multi-view stereo system. Their follow-up work [Hu et al. 2014b] uses only a single RGB-D sensor (Kinect) as acquisition hardware and addresses the problem of constrained hairstyle digitization such as braids using a database of procedurally generated examples. Our method only requires a single reference photo and a few 2D strokes as input, and is thus more cost-effective, deployable, and user-friendly. Existing sketch-based hair modeling techniques [Wither et al. 2007; Fu et al. 2007; Yu et al. 2014] are purely manual design tools since they do not take any reference photographs as direct input to the algorithms. State-of-the-art image-based hair modeling methods [Chai et al. 2012; Chai et al. 2013] can handle a single-view image but require high-quality photographs with complete hair strands and a frontal face as input. Our approach of using a hairstyle database can handle challenging hair photographs that are largely occluded, as well as these with different head poses.

**Single-view reconstruction.** Several algorithms have been proposed to estimate facial models [Blanz and Vetter 1999], hands [Wang and Popović 2009], and hair appearance [Bonneel et al. 2009] from a single-view input image or video. To address this ill-conditioned problem, most of these methods rely on a reduced parametric models that encode the shape/appearance variations of the specific target phenomenon to achieve a reasonable model estimation. Unfortunately, such representation is difficult to obtain for hair due to the vast diversity in the structure and the style.

**Modeling from a large database.** Following the seminal work of [Funkhouser et al. 2004], several important advances on data-driven modeling have been introduced in recent years [Chaudhuri et al. 2011; Xu et al. 2011; Xu et al. 2012; Kalogerakis et al. 2012; Shen et al. 2012; Kholgade et al. 2014]. The use of an existing shape database can significantly speed up the modeling process compared to building a new shape from scratch. For example, Kalogerakis et al. [2012] have proposed a method to synthesize new shapes by assembling individual parts from a shape repository. We adopt a similar model assembly-based approach to the problem of hair digitization. Unlike many existing data-driven techniques, our approach does not require pre-segmentation of examples in the database, which is difficult to achieve for most hairstyles due to their continuous shape distribution and indiscernible components.

**Sketch-based modeling.** Intuitive 2D sketches are popular user-interfaces for prototyping designs and scalable content creation. We refer the readers to [Olsen et al. 2009] for a comprehensive survey of earlier work about sketch-based modeling interfaces. Some recent work has been applied to the modeling of photo compositions [Chen et al. 2009], quad meshes [Takayama et al. 2013], indoor scenes [Xu et al. 2013], 3D curve networks [Xu et al. 2014a], and part assemblies [Huang et al. 2014]. Recently Cherin et al. [2014] proposed a method to fit a single 2D sketch with a piecewise 3D helix curve. Our algorithm for the modeling of individual strand shape (Section 5) is inspired from this technique but also incorporates the interaction of neighboring hair strands.



**Figure 2:** Preprocessing of an example hairstyle. From left to right: (a) the original mesh (downloaded from online repositories and roughly aligned with a given head model); (b) hair strands by uniformly sampling the original mesh; (c) the final example by growing hair strands from the scalp.

### 3 Database Construction

We construct a database of hairstyles spanning a wide range of overall shapes and different local details by collecting 343 examples from several repositories available online [Electronic Arts 2014; Newsea 2014]. Most of these hairstyles are manually created by various artists and gamers. The original data of each hairstyle is specified as a set of triangle meshes, while each mesh represents a wisp of hair strands with consistent orientation and length.

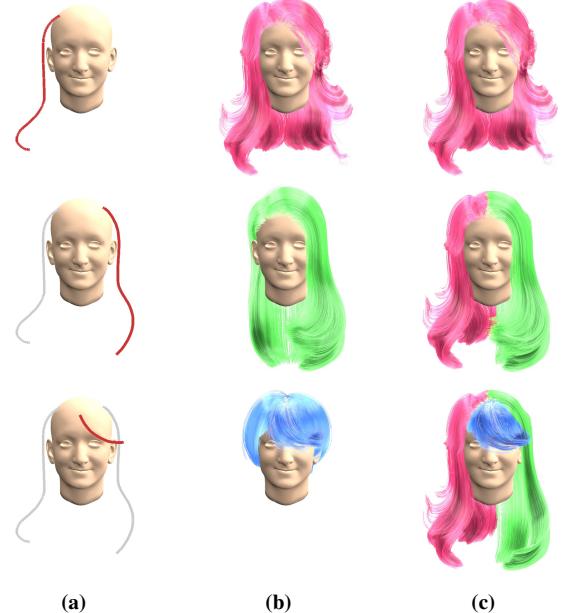
Since many of the original hairstyle models are created under different poses, we manually transform the global pose of each hairstyle to roughly align them with a fixed standard head model by translating, rotating and/or scaling. To convert the mesh into hair strands for later computation, we first uniformly sample all the meshes in the hairstyle following the local orientation and obtain a set of hair strands, where each strand is represented as a set of equally spaced sample points. Next we build a 3D orientation field from the hair strands using the method of Wang et al. [2009] and smoothly diffuse the field into the entire 3D volume as proposed by Paris et al. [2008]. Guided by the diffused 3D orientation field, we grow hair strands from 10000 uniformly distributed roots on the scalp and consider these strands as an example hairstyle  $\mathcal{E}$  in the database  $\mathcal{D}$ . See Figure 2 for a concrete example. We finally augment the database by flipping each example w.r.t. the plane of reflection symmetry of the head model and obtain a database containing 686 examples in total.

**Notations.** Throughout this paper we use the italic font to denote a scalar or a single entity (e.g., a sample point  $s$  on a hair strand), the bold font for a vector (e.g., a position  $\mathbf{p}$  in 3D space), and the calligraphic font for a set (e.g., the database  $\mathcal{D}$  as a set of hairstyles, a hairstyle  $\mathcal{H}$  as a set of strands, and a strand  $\mathcal{S}$  as a set of samples).

### 4 Modeling Large-Scale Structure

Given a reference photograph of the 3D target hairstyle, we first determine a transformation matrix  $\mathbf{T}$  to align the standard head model with the head pose in the photo. For photographs of frontal faces, the transformation matrix can be automatically computed based on the detection of a set of 2D facial feature points [Baltrušaitis et al. 2013]. For other photos we manually align the head to estimate the transformation matrix approximately based on visual observation.

Next, we let the user draw a few 2D strokes over the reference photo to guide the modeling of the large-scale structure. For each stroke, we search for the best matching example hairstyle in the database



**Figure 3:** Retrieving and combining multiple hairstyles. From left to right: (a) user strokes with the latest one shown with red; (b) the best matching example hairstyle retrieved from the database based on the red stroke; (c) the combined hairstyle at the current step, with each hair sample colorized according to the label of the grid which contains the sample point.

(Section 4.1). We then combine all the retrieved example hairstyles together to form a consistent large-scale structure of the 3D target hairstyle (Section 4.2). In addition, we provide some optional editing operations to refine the modeling result of the large-scale structure (Section 4.3).

#### 4.1 Example Retrieval

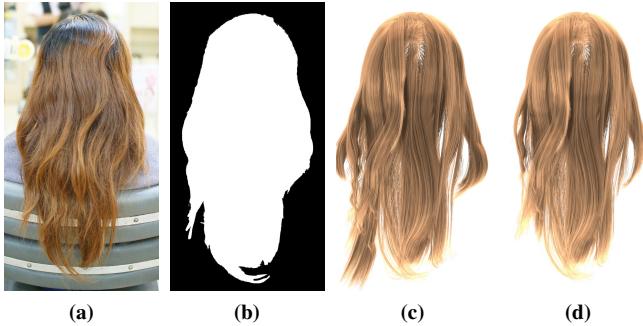
For many reference photos, it is difficult or nearly impossible to automatically extract the complete structure of hair strands due to the complexity of the hairstyles, self-occlusions between hair strands, and the conditions under which the photos were taken. As a result, we ask the user to draw several 2D strokes  $\{\mathcal{U}\}$  based on his/her observation of the reference photograph. Each stroke is required to be from the root to the tip following the orientation of hair strands in the photograph. We consider these user strokes as essential structures of the target hairstyle projected onto the image plane, which are necessary to reveal the full hair connectivity and topology.

To measure the difference between a 2D user stroke  $\mathcal{U}$  and a 3D hair strand  $\mathcal{S}$ , we project the strand onto the image plane using the transformation matrix  $\mathbf{T}$  mentioned above. We find the closest sample  $s_j$  on the projected strand for each sample  $s_i$  on the user stroke  $\mathcal{U}$  and compute the difference as:

$$D(\mathcal{U}, \mathcal{S}) = \sum_{s_i \in \mathcal{U}} \min_{s_j \in \mathcal{S}} |\mathbf{p}(s_i) - \mathbf{p}(s_j)| \quad (1)$$

where  $|\mathbf{p}(s_i) - \mathbf{p}(s_j)|$  is the distance between the positions of  $s_i$  and  $s_j$  on the image plane.

We define the difference between a user stroke  $\mathcal{U}$  and an example hairstyle  $\mathcal{E}$  in the database by simply searching for the hair strand  $\mathcal{S}$  of the hairstyle that minimizes the stroke-strand difference as



**Figure 4:** Editing the structure of a hairstyle with a cutting tool. From left to right: (a) reference photo; (b) manually extracted binary mask; (c) the hair structure before editing; (d) the hair structure after editing. Original image courtesy of Eu Hairdresser.

defined in Equation 1:

$$D(\mathcal{U}, \mathcal{E}) = D(\mathcal{U}, \{\mathcal{S}_i\}) = \min_{\mathcal{S}_i \in \mathcal{E}} D(\mathcal{U}, \mathcal{S}_i) \quad (2)$$

For each user stroke  $\mathcal{U}_i$ , we search for the best matching example hairstyle  $\mathcal{E}_i$  from the database  $\mathcal{D}$  which minimizes the stroke-hairstyle difference defined in Equation 2. We also store the best matching hair strand  $\mathcal{S}_i \in \mathcal{E}_i$  which minimizes the stroke-strand difference among all the strands in  $\mathcal{E}_i$  for the computation in Section 4.2.

**Discussion.** We have tried several different metrics for the retrieval of the best matching example, including an alternative way based on local curvatures. We have found that the simple form in Equation 1 to be highly effective and sufficiently robust since we focus on large-scale structures rather than local shape details during this retrieval step.

## 4.2 Hairstyle Combination

After the example retrieval step as described above, our next goal is to combine these retrieved hairstyles into a consistent large-scale structure representing the target hairstyle. The combined hairstyle should follow the guidance of the user strokes  $\{\mathcal{U}_i\}$  and maintain local consistency while the retrieved example hairstyles  $\{\mathcal{E}_i\}$  may be significantly different from each other. Due to the diversity of the retrieved hairstyles  $\{\mathcal{E}_i\}$ , it is usually difficult to directly combine or blend different hair strands together. As a result, we build a 3D orientation field  $\mathbf{F}_i$  for each retrieved hairstyle  $\mathcal{E}_i$  [Wang et al. 2009], and perform the combination over the grids of these fields.

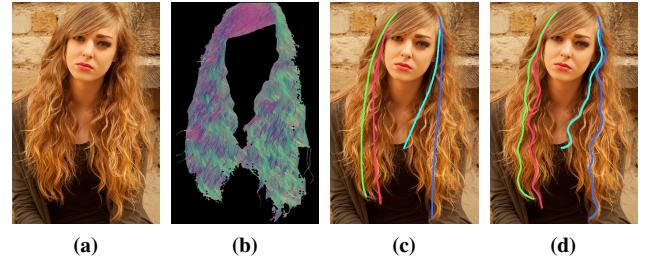
We formulate the task as a multi-label assignment problem. Specifically, we consider each retrieved example hairstyle  $\mathcal{E}_i$  together with the best matching hair strand  $\mathcal{S}_i \in \mathcal{E}_i$  as a label  $l_i$ , and try to assign the optimal label for each grid  $g$  in the 3D space by minimizing the following energy function:

$$E(\{g_i\}, \{l_i\}) = \sum_i E_d(g_i, l_i) + \sum_{i, g_j \in \mathcal{N}(g_i)} E_s(g_i, g_j, l_i, l_j) \quad (3)$$

where the first term is a *fitness* term to ensure that the combined orientation field follows the guidance of the user strokes, and is computed as the minimum distance between the grid center and the best matching strand  $\mathcal{S}_{l_i}$  corresponding to the label  $l_i$ :

$$E_d(g_i, l_i) = \min_{s_j \in \mathcal{S}_{l_i}} |\mathbf{p}(g_i) - \mathbf{p}(s_j)| \quad (4)$$

where  $\mathbf{p}(g_i)$  and  $\mathbf{p}(s_j)$  are the positions of the grid center  $g_i$  and the sample  $s_j$  respectively. The second term in Equation 3 is a



**Figure 5:** Illustration of 2D deformation. From left to right: (a) reference photo; (b) 2D orientation map; (c) four hair strands before deformation; (d) the corresponding hair strands after deformation (100 iterations). Original image courtesy of Gokhan Altintas.

smoothness term to ensure that every pair of adjacent grids (i.e.,  $g_j$  belongs to the neighborhood  $\mathcal{N}(g_i)$  of  $g_i$ ) are assigned to labels with consistent local orientations:

$$E_s(g_i, g_j, l_i, l_j) = \begin{cases} 0 & \text{if } l_i = l_j \\ 5 & \text{elif } l_i, l_j \text{ compatible} \\ 100 & \text{otherwise} \end{cases} \quad (5)$$

In Equation 5, we consider two different labels  $l_i$  and  $l_j$  to be compatible for two adjacent grids  $g_i$  and  $g_j$ , if and only if the corresponding orientation fields have similar local orientations  $\mathbf{o}_i = \mathbf{F}_i(g_i)$  and  $\mathbf{o}_j = \mathbf{F}_j(g_j)$ , i.e., the dot product between  $\mathbf{o}_i$  and  $\mathbf{o}_j$  is larger than a threshold  $\tau = 0.7$ .

The energy function defined in Equation 3 can be minimized via the graph-cuts algorithm as described in Delong et al. [2012]. After computing the optimal label for each grid, we obtain a combined orientation field  $\mathbf{F}_c$  where the local orientation is specified according to the assigned label for each grid. We then grow the hair strands from the combined orientation field  $\mathbf{F}_c$  to obtain the combined hairstyle  $\mathcal{H}_c$  as the large-scale structure of the target hairstyle. See Figure 3 for an illustration about combining multiple hairstyles.

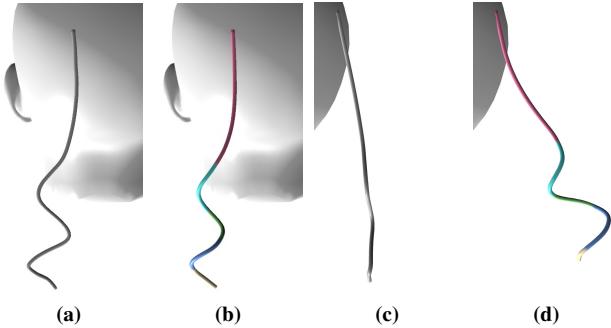
**Discussion.** Instead of combining multiple hairstyles through intermediate 3D grids, an alternative way is to directly select hair strands from those retrieved hairstyles by considering the unified set of hair roots as the domain to perform the combination. However, we have found that it is difficult for this approach to ensure local consistency across the entire 3D volume except around the hair roots.

## 4.3 Structure Editing

Once we obtain the combined hairstyle  $\mathcal{H}_c$ , we can optionally perform editing operations on the large-scale structure. In particular, we have implemented a cutting tool to manipulate the contour of  $\mathcal{H}_c$  and introduce some random variations to the length of hair strands in  $\mathcal{H}_c$ , following the method introduced by Chai et al. [2013]. See Figure 4 for an example of editing the contour with a 2D mask which is manually prepared from the reference photo.

## 5 Modeling Strand Shapes

After obtaining the large-scale structure  $\mathcal{H}_c$  of the target hairstyle, we refine the shape of each individual strand in  $\mathcal{H}_c$  such that (1) the projected 2D local details of the modeling result are as close as possible to the reference photo (Section 5.1); (2) the shapes of the 3D strands are physically plausible (Section 5.2); and (3) the global structure of the hairstyle is preserved while maintaining the local coherency between neighboring strands (Section 5.3).



**Figure 6:** Fit with piecewise helix curves. From left to right: (a) hair strand before fitting; (b) the fitted piecewise helix curve with each segment visualized using a different color; (c) & (d) another view of (a) & (b).

## 5.1 2D Orientations

We compute a 2D orientation map  $\mathbf{M}$  from the reference photo using the method described by Luo et al. [2013]. Then we deform each hair strand  $\mathcal{S} \in \mathcal{H}_c$  based on the 2D orientation map as well as the visibility of each hair strand in the view of the reference photo. We consider a hair strand as visible if and only if its root is visible from current point of view. We found this simple heuristic to work well in practice for all of our examples. Our key idea to enforce the similarity of 2D orientations is to gradually deform each visible hair strand with 100 iterations so that for each sample  $s_i$  on the strand  $\mathcal{S}$ , the projected local orientation  $\mathbf{o}(s_i) = \mathbf{p}(s_{i+1}) - \mathbf{p}(s_i)$  is as close as possible to the local orientation  $\mathbf{M}(s_i)$  on the 2D map.

In each iteration, every hair strand  $\mathcal{S}$  is projected onto the image plane based on the viewpoint of the photograph. We then deform the strand  $\mathcal{S}$  according to the 2D orientation map by minimizing the following energy:

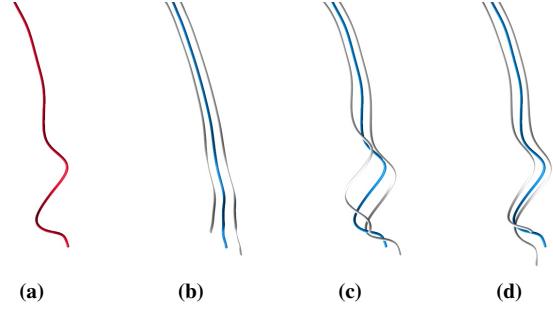
$$E(\{\mathbf{p}(s_i)\}) = \sum_i |\mathbf{p}(s_i) - \bar{\mathbf{p}}(s_i)|^2 + \alpha_1 |\mathbf{o}(s_i) - \mathbf{M}(s_i)|^2 + \alpha_2 |\mathbf{p}(s_{i-1}) - 2\mathbf{p}(s_i) + \mathbf{p}(s_{i+1})|^2 \quad (6)$$

where  $\mathbf{p}(s_i)$  is the projected 2D position of sample  $s_i$ ,  $\bar{\mathbf{p}}(s_i)$  the initial position of sample  $s_i$  before the deformation,  $s_{i-1}$  and  $s_{i+1}$  the predecessor and successor of  $s_i$  on the same strand, and  $\delta$  the average length between two adjacent samples on the strand. The constants  $\alpha_1$  and  $\alpha_2$  are specified to control the relative weights of the orientation and curvature terms with respect to the first position term. We set  $\alpha_1 = 10$  and  $\alpha_2 = 10$  for all the results in our paper. We solve the linear system about all the sample positions  $\{\mathbf{p}(s_i)\}$  to minimize Equation 6 and obtain the deformed shape of strand  $\mathcal{S}$ . See Figure 5 for an illustration about the deformation.

## 5.2 Depth estimation

The reference photo does not provide any depth information for the hair strands, and the 3D strand shapes after the deformation step described above may not be physically plausible. In theory, when there is no external force like gravity or friction, the 3D shape of each hair strand will be a helix whose curvature and torsion are constants. However, in reality most of the hair strands are influenced by the global gravity and hair-hair interactions. Furthermore, the curvature of the helix does not remain invariant under 2D projections. As a result, it is impossible to accurately depict the curvature and torsion values solely from the 2D projection of a hair strand.

Inspired by the fact that a hair strand can be modeled as piecewise helices [Bertails et al. 2006], to estimate the depth information of



**Figure 7:** Linear blending skinning for a cluster of hair strands. From left to right: (a) the deformed center strand  $\mathcal{S}_d$ ; (b) strands in the same cluster before deformation with the center strand  $\mathcal{S}_c$  highlighted in blue; (c) the naïve deformation result by directly transferring the per-sample offset; (d) our deformation result via linear blend skinning.

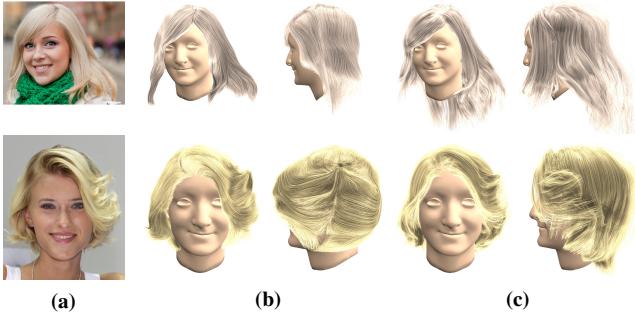
a given strand  $\mathcal{S}$ , we fit its projected 2D shape  $\mathcal{S}_p$  on the image plane with a piecewise helix curve  $\mathcal{S}_f$ , using the method proposed by Cherin et al. [2014] (see Figure 6). To determine the position of the fitted strand shape in the global coordinate system, we translate the piecewise helix curve  $\mathcal{S}_f$  to align the first sample to the original position before the fitting.

## 5.3 3D Shapes

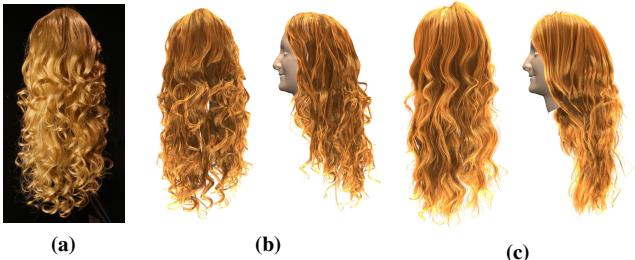
To apply the estimated depth information to the entire hairstyle, we first group all the hair strands into 300 clusters via  $k$ -means clustering based on the root positions and strand shapes as in Wang et al. [2009]. For each cluster, we apply the helix fitting algorithm described above to the center strand  $\mathcal{S}_c$  and get the fitted piecewise helix curve  $\mathcal{S}_f$ . To combine the local details of  $\mathcal{S}_f$  with the overall structure of  $\mathcal{S}_c$ , we align  $\mathcal{S}_f$  with  $\mathcal{S}_c$  using 40 iterations of non-rigid ICP [Li et al. 2009] and obtain a deformed strand  $\mathcal{S}_d$ .

Next, we deform the other strands in the same cluster as  $\mathcal{S}_c$  consistently. One possible way is to directly transfer the per-sample offset between  $\mathcal{S}_c$  and  $\mathcal{S}_d$  to each strand in the cluster. But this naïve approach cannot generate coherent deformation result, since it does not take into account the global pose of the strand cluster. Consequently, we adopt a linear blend skinning approach [Lewis et al. 2000] to transfer the deformation between  $\mathcal{S}_c$  and  $\mathcal{S}_d$  to other strands. Specifically, we consider  $\mathcal{S}_c$  as the rest pose shape and  $\mathcal{S}_d$  as the deformed shape. We use each segment between two adjacent samples on both  $\mathcal{S}_c$  and  $\mathcal{S}_d$  as a bone. For each bone  $\mathbf{B}_c$  in  $\mathcal{S}_c$ , we compute the twist-free material frame [Bergou et al. 2008], and measure the transformation between  $\mathbf{B}_c$  and the corresponding bone  $\mathbf{B}_d$  in  $\mathcal{S}_d$  within the material frame. For each sample  $s$  on other strands in the same cluster, we search for the four closest bones of  $\mathcal{S}_c$  and compute the skinning weights based on the distances. Then we blend the transformations of those bones linearly according to the weights and apply the blended transformation to get the deformed position for  $s$ . See Figure 7 for a comparison of deformation results between a naïve offset transfer method (Figure 7c) and the linear blend skinning technique (Figure 7d).

**Discussion.** [Chai et al. 2014] also used a skinning model to interpolate the motion of a full set of hair strands from simulated guide strands. They estimated the skinning weights from sequences of precomputed simulation data by solving a constrained least-squares problem which ensures the interpolation match the training data closely. In our case of modeling static hair shapes, we find it sufficient to estimate skinning weights based on spatial distances.



**Figure 8:** Comparison with state-of-the-art sketch-based hair modeling method. From left to right: (a) reference photos; (b) results by [Chai et al. 2013]; (c) our results. From top to bottom, original images courtesy of Chris Zerbes and Georg Sander.

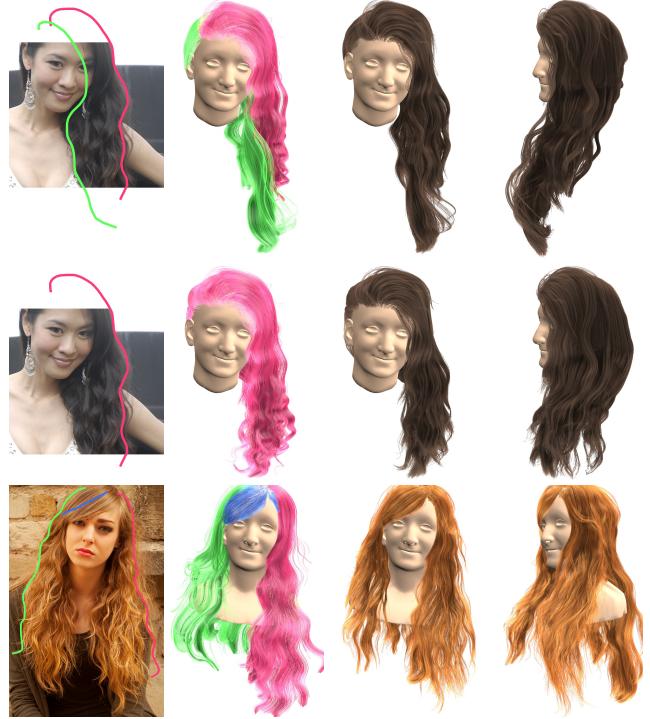


**Figure 9:** Comparison with 3D hair capture. From left to right: (a) reference photo; (b) the reconstruction result of [Hu et al. 2014a] based on capturing the target hairstyle from 50 different views; (c) our result from the single view of (a).

## 6 Results

**Hairstyles.** To demonstrate the generality and effectiveness of our method, we experiment with a variety of examples from single-view photographs, as shown in Figure 1 and 13. Specifically, we can handle different overall shapes (including both long and short hair strands), different curliness (ranging from straight strands to very curly ones) and different head poses (including frontal, profile and back views). Thanks to the hairstyle database, we can generate modeling results with complete hair structures even when the input photos are cropped and largely incomplete (Figure 1, the second row in Figure 13). The user only needs to provide a few 2D strokes using the input photo as reference. Our method then automatically generates plausible 3D hair strands of the target hairstyle.

**Comparisons.** We compare our approach with the state-of-the-art sketch-based single-view hair modeling method [Chai et al. 2013]. Their method relies on information provided by the input photo and some heuristics for depth estimation which only work well for images from the frontal view. As a result, their method cannot handle challenging cases when the hair/head are partially missing (e.g., Figure 1) or the photos are not taken from the frontal view (e.g., the last three rows in Figure 13). By leveraging the prior knowledge of our 3D hairstyle database, we are able to generate a reasonable and complete structure of the target hairstyle even when the reference photo is incomplete. Furthermore, our strand synthesis algorithm based on piecewise helix fitting and linear blend skinning can produce more natural and faithful 3D strand shapes compared to existing techniques as shown in Figure 8.

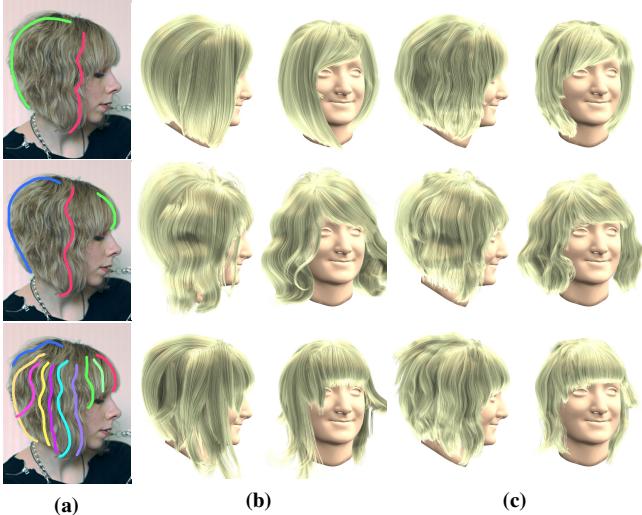


**Figure 10:** Modeling results for the same reference photos from reduced numbers of strokes. From left to right: the user strokes; colored visualization of the hairstyle combination result; final hair strands from two different views.

We also compare our approach with cutting-edge 3D hair capture method [Hu et al. 2014a]. As shown in Figure 9, the overall quality of our output is comparable to theirs, even though our approach uses much less information as input (a single reference photo as in Figure 9a versus a dense 3D point cloud obtained from 50 images). The user strokes for Figure 8 and 9 are shown in the accompanying video.

**Evaluations.** To assess the robustness of our method, we produce several modeling results using different sets of user strokes on the same reference photo. As shown in Figure 10, our approach can generate reasonable modeling results even when very few strokes are provided (compared to the corresponding results in Figure 1 and 13). Intuitively, the modeling results are faithful when more strokes are sketched. In Figure 11, we collected different sets of strokes from three users to guide the modeling process from the same reference photo of a profile view. The large-scale structures of the modeling results from different user strokes can be quite different, because the retrieved example hairstyles can vary significantly. However, all three results in Figure 11 are visually reasonable solutions and closely match the local details of the reference photo.

**Limitations.** Figure 12 shows two failure cases of our current system. Due to the lack of appropriate examples in the database, we cannot handle extreme cases such as when the hair in the reference photo is much longer than all other example hairstyles, or the head pose is so tilted that the overall shape of the target hairstyle no longer matches any of those in the database. It is possible to augment the solution space of the database to cover such cases by adding more examples. Another potential remedy is to grow and/or simulate these examples online based on the reference photo.



**Figure 11:** Modeling results for the same reference photo using different sets of user strokes. From left to right: (a) the user strokes; (b) combined hairstyles visualized from two different views; (c) final hair strands visualized from two different views.

**Implementation and statistics.** The parameters mentioned in earlier sections work for all the cases when we measure 2D variables in pixels and 3D variables in millimeters. The grid size is set to be 2 mm for all the intermediate orientation fields, except in the example retrieval stage we use a larger grid size of 8 mm to support interactive preview of the hairstyle combination result when the user draws strokes. Performance-wise, our system takes about 20 minutes to create a single 3D result (with a 3.5GHz Intel Core i7 and 16GB RAM). Our sketch-based example retrieval algorithm runs interactively and the user usually draws all the strokes within a few seconds. The hairstyle combination step can be finished within one minute. Manual segmentation of the hair part from the input photo takes no more than five minutes (in Adobe Photoshop with the “Polygonal Lasso” and “Refine Edge” tools). The 2D deformation step is the most time-consuming part in our pipeline since we iteratively deform each individual strand which takes about ten minutes for each result of 10000 strands. The final strand synthesis step takes about two minutes for the 300 clusters of strands.

## 7 Conclusion

We introduced a data-driven framework for modeling realistic and complex 3D hairstyles from a single-view input photograph taken in the wild. We showed that a database containing complete example hairstyles can be used to model a wide range of target hairstyles using a novel assembly-based approach. Our method only requires a few input user strokes and can handle highly incomplete and low quality reference photos. We also ensure physically plausible strand shapes in the output data using a new strand synthesis algorithm based on piecewise helix fitting and linear blend skinning.

As a data-driven method, the effectiveness of our method is mainly determined by the richness of our hairstyle database. In particular, our current system cannot handle constrained hairstyles like braids [Hu et al. 2014b], due to the lack of related examples in the database to infer the desired structures. We plan to augment the database with such examples and to generalize our representation to handle more complicated cases. Furthermore, the fine scale structures of our output strands highly depend on the quality of the reference photos. The ability to extract more accurate and complex 2D features from photographs can significantly improve the reliability and quality of



**Figure 12:** Failure cases. Original image courtesy of Devon Christopher Adams and Maria Morri.

our modeling results.

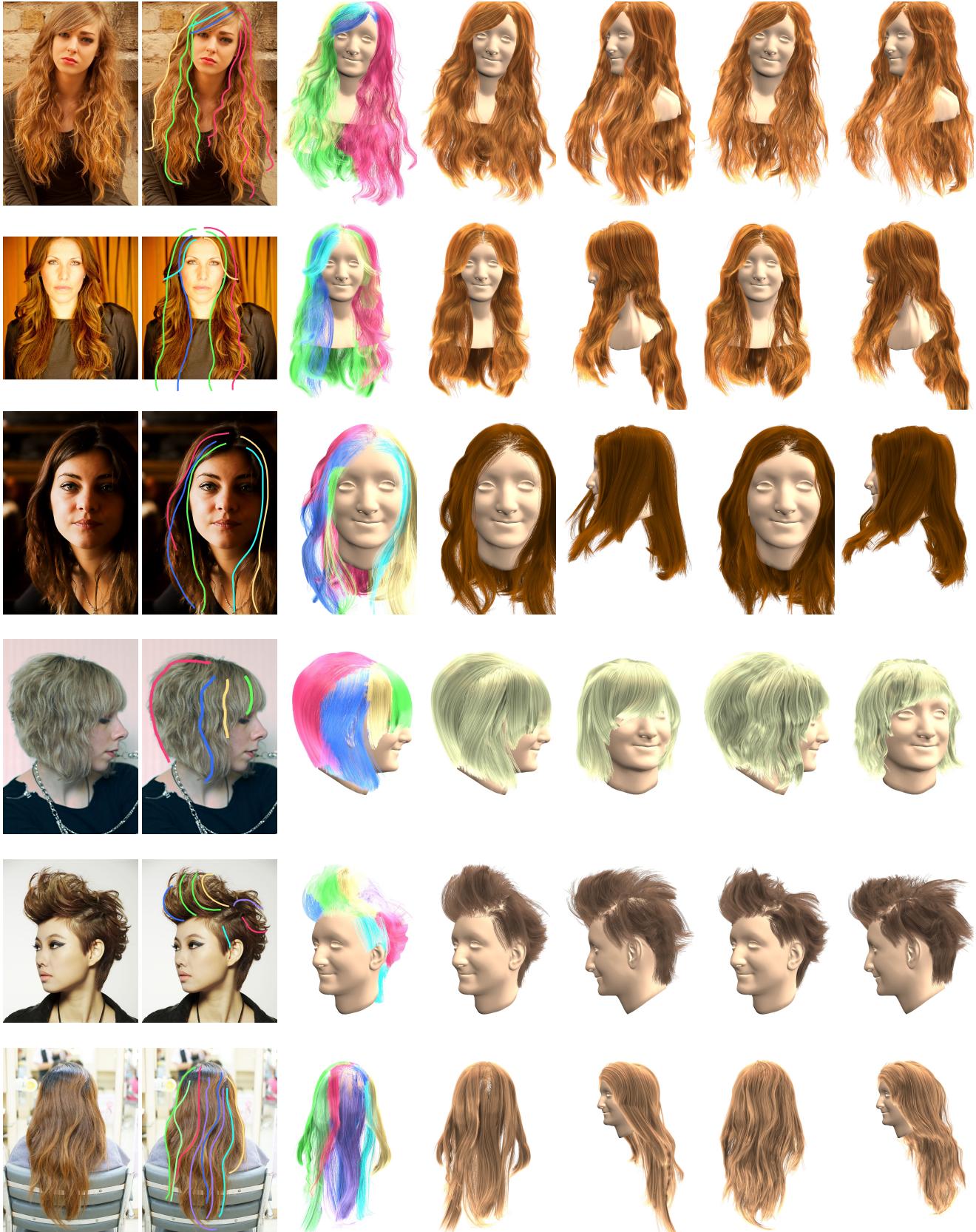
While our current effort focuses on the geometry of hairstyles, i.e., the 3D structure and shapes of hair strands, we are interested in exploring appearance estimation techniques of hair strands, especially from a single reference photograph. Even though this work is designed to handle single-view photographs, we believe our method can be extended to produce a consistent hair model from multiple-view inputs. Finally, when constructing our database, we unify all the example hairstyles with a neutral head model. In reality the scalp shapes from where the hair strands grow can vary significantly across different people, which is another interesting future direction is to investigate.

## Acknowledgements

We would like to thank the gamers and artists for making their hair models available on The Sims Resource and Newsea platform; Lingyu Wei for the script to retrieve the data; the photo owners for sharing their work on Flickr; Menglei Chai and Kun Zhou for producing the comparisons; Steve Marschner for the multi-view stereo data; Justin Solomon for the helpful discussions; Sema Berkmen for sharing her head model; Szymon Rusinkiewicz for the rendering code; Kyle Olszewski and Jessica Rossky for proofreading. This research is supported by grants from Oculus/Facebook, USC’s Integrated Media Systems Center (IMSC), Adobe Systems, Pelican Imaging, and the Google Research Faculty Award.

## References

- BALTRUSAITIS, T., ROBINSON, P., AND MORENCY, L.-P. 2013. Constrained local neural fields for robust facial landmark detection in the wild. In *IEEE ICCVW*, 354–361.
- 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., CANI, M.-P., QUERLEUX, B., LEROY, F., AND LÉVÈQUE, J.-L. 2006. Super-helices for predicting the dynamics of natural hair. *ACM Trans. Graph.* 25, 3, 1180–1187.
- BLANZ, V., AND VETTER, T. 1999. A morphable model for the synthesis of 3d faces. In *SIGGRAPH ’99*, 187–194.
- BONNEEL, N., PARIS, S., VAN DE PANNE, M., DURAND, F., AND DRETTAKIS, G. 2009. Single photo estimation of hair appearance. In *EGSR’09*, 1171–1180.
- 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.



**Figure 13:** Single-view modeling results of various hairstyles from reference photographs. From left to right: the reference photo; user strokes; colored visualization of the hairstyle combination result; combined hairstyle from two different views; final hair strands from two different views. From top to bottom, original images courtesy of Gokhan Altintas, Kris Krüg, Rivodoza, Maria Morri, Wallace Lan, and Eu Hairdresser.

- CHAI, M., ZHENG, C., AND ZHOU, K. 2014. A reduced model for interactive hairs. *ACM Trans. Graph.* 33, 4, 124:1–124:11.
- CHAUDHURI, S., KALOGERAKIS, E., GUIBAS, L., AND KOLTUN, V. 2011. Probabilistic reasoning for assembly-based 3D modeling. *ACM Trans. Graphics* 30, 4, 35:1–35:10.
- CHEN, T., CHENG, M.-M., TAN, P., SHAMIR, A., AND HU, S.-M. 2009. Sketch2photo: Internet image montage. *ACM Trans. Graph.* 28, 5, 124:1–124:10.
- CHERIN, N., CORDIER, F., AND MELKEMI, M. 2014. Modeling piecewise helix curves from 2d sketches. *Computer-Aided Design* 46, 258–262.
- CHOE, B., AND KO, H.-S. 2005. A statistical wisp model and pseudophysical approaches for interactive hairstyle generation. *IEEE Trans. Vis. Comput. Graph.* 11, 2, 160–170.
- DELONG, A., OSOKIN, A., ISACK, H. N., AND BOYKOV, Y. 2012. Fast approximate energy minimization with label costs. *International Journal of Computer Vision* 96, 1, 1–27.
- DEROUET-JOURDAN, A., BERTAILS-DESCOUBES, F., DAVIET, G., AND THOLLON, J. 2013. Inverse dynamic hair modeling with frictional contact. *ACM Trans. Graph.* 32, 6, 159:1–159:10.
- ECEVARRIA, J. I., BRADLEY, D., GUTIERREZ, D., AND BEELER, T. 2014. Capturing and stylizing hair for 3d fabrication. *ACM Trans. Graph.* 33, 4, 125:1–125:11.
- ELECTRONIC ARTS, 2014. The Sims Resource. <http://www.thesimsresource.com/>.
- FU, H., WEI, Y., TAI, C.-L., AND QUAN, L. 2007. Sketching hairstyles. In *SBIM '07*, 31–36.
- FUNKHOUSER, T., KAZHDAN, M., SHILANE, P., MIN, P., KIEFER, W., TAL, A., RUSINKIEWICZ, S., AND DOBKIN, D. 2004. Modeling by example. *ACM Trans. Graphics* 23, 3, 652–663.
- HU, L., MA, C., LUO, L., AND LI, H. 2014. Robust hair capture using simulated examples. *ACM Trans. Graph.* 33, 4, 126:1–126:10.
- HU, L., MA, C., LUO, L., WEI, L.-Y., AND LI, H. 2014. Capturing braided hairstyles. *ACM Trans. Graph.* 33, 6, 225:1–225:9.
- HUANG, Z., FU, H., AND LAU, R. W. H. 2014. Data-driven segmentation and labeling of freehand sketches. *ACM Trans. Graph.* 33, 6, 175:1–175:10.
- JAKOB, W., MOON, J. T., AND MARSCHNER, S. 2009. Capturing hair assemblies fiber by fiber. *ACM Trans. Graph.* 28, 5, 164:1–164:9.
- KALOGERAKIS, E., CHAUDHURI, S., KOLLER, D., AND KOLTUN, V. 2012. A probabilistic model for component-based shape synthesis. *ACM Trans. Graph.* 31, 4, 55:1–55:11.
- KHOLGADE, N., SIMON, T., EFROS, A., AND SHEIKH, Y. 2014. 3d object manipulation in a single photograph using stock 3d models. *ACM Trans. Graph.* 33, 4, 127:1–127:12.
- 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.
- LEWIS, J. P., CORDNER, M., AND FONG, N. 2000. Pose space deformation: A unified approach to shape interpolation and skeleton-driven deformation. In *SIGGRAPH '00*, 165–172.
- LI, H., ADAMS, B., GUIBAS, L. J., AND PAULY, M. 2009. Robust single-view geometry and motion reconstruction. *ACM Trans. Graph.* 28, 5, 175:1–175:10.
- LUO, L., LI, H., AND RUSINKIEWICZ, S. 2013. Structure-aware hair capture. *ACM Trans. Graph.* 32, 4, 76:1–76:12.
- NEWSEA, 2014. Newsea SIMS. <http://www.newseasims.com/>.
- OLSEN, L., SAMAVATI, F. F., SOUSA, M. C., AND JORGE, J. A. 2009. Sketch-based modeling: A survey. *Computers & Graphics* 33, 1, 85–103.
- 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.
- 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.
- SHOTTON, J., FITZGIBBON, A., COOK, M., SHARP, T., FINOCCHIO, M., MOORE, R., KIPMAN, A., AND BLAKE, A. 2011. Real-time human pose recognition in parts from single depth images. In *CVPR '11*, 1297–1304.
- TAKAYAMA, K., PANZZO, D., SORKINE-HORNUNG, A., AND SORKINE-HORNUNG, O. 2013. Sketch-based generation and editing of quad meshes. *ACM Trans. Graph.* 32, 4, 97:1–97:8.
- WANG, R. Y., AND POPOVIĆ, J. 2009. Real-time hand-tracking with a color glove. *ACM Trans. Graph.* 28, 3, 63:1–63:8.
- 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., BERTAILS, F., KIM, T.-Y., MARSCHNER, S. R., CANI, M.-P., AND LIN, M. C. 2007. A survey on hair modeling: Styling, simulation, and rendering. *IEEE TVCG* 13, 2, 213–234.
- WENG, Y., WANG, L., LI, X., CHAI, M., AND ZHOU, K. 2013. Hair interpolation for portrait morphing. *Computer Graphics Forum* 32, 7, 79–84.
- 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.
- XU, K., ZHANG, H., COHEN-OR, D., AND CHEN, B. 2012. Fit and diverse: Set evolution for inspiring 3d shape galleries. *ACM Trans. Graph.* 31, 4, 57:1–57:10.
- XU, K., CHEN, K., FU, H., SUN, W.-L., AND HU, S.-M. 2013. Sketch2scene: Sketch-based co-retrieval and co-placement of 3d models. *ACM Trans. Graph.* 32, 4, 123:1–123:15.
- XU, B., CHANG, W., SHEFFER, A., BOUSSEAU, A., MCCRAE, J., AND SINGH, K. 2014. True2form: 3d curve networks from 2d sketches via selective regularization. *ACM Trans. Graph.* 33, 4, 131:1–131:13.
- XU, Z., WU, H.-T., WANG, L., ZHENG, C., TONG, X., AND QI, Y. 2014. Dynamic hair capture using spacetime optimization. *ACM Trans. Graph.* 33, 6, 224:1–224:11.
- YU, X., YU, Z., CHEN, X., AND YU, J. 2014. A hybrid image-cad based system for modeling realistic hairstyles. In *I3D '14*, 63–70.
- YUKSEL, C., SCHAEFER, S., AND KEYSER, J. 2009. Hair meshes. *ACM Trans. Graph.* 28, 5, 166:1–166:7.
- ZHOU, S., FU, H., LIU, L., COHEN-OR, D., AND HAN, X. 2010. Parametric reshaping of human bodies in images. *ACM Trans. Graph.* 29, 4, 126:1–126:10.